Algebraic Connectivity of Graphs, with Applications

YounHun Kim

Department of Computer Science, Brown University

Advisor: Prof. Sorin Istrail

Second Reader: Prof. Eli Upfal

Submission for Math - Computer Science joint concentration

Spring 2016

Abstract

The analysis of matrices associated with discrete, pairwise comparisons can be a very useful toolbox for a computer scientist. In particular, Spectral Graph Theory is based on the observation that eigenvalues and eigenvectors of these matrices betray a lot of properties of graphs associated with them. The first major section of this paper is a survey of key results in Spectral Graph Theory. There are fascinating results involving the connectivity, spanning trees, and a natural measure of bi-partiteness in graphs. Some common applications include clustering, graph cuts and random walks. In this study, we explore Spectral Graph Theory and possible ways to use these concepts in other areas. One problem that we will address is the Haplotype Phasing problem from Computational Biology, using spectral methods to address certain issues with Clark's phasing method.

Contents

1 Introduction

The analysis of matrices associated with discrete, pairwise comparisons can be a very useful toolbox for a computer scientist. In particular, Spectral Graph Theory is based on the observation that eigenvalues and eigenvectors of these matrices betray a lot of properties of graphs associated with them. The first major section of this paper is a survey of key results in Spectral Graph Theory. There are fascinating results involving the connectivity, spanning trees, and a natural measure of bi-partiteness in graphs.

Some common applications include clustering, graph cuts and random walks. In this study, we explore Spectral Graph Theory and possible ways to use these concepts in other areas. One problem that we will address is the Haplotype Phasing problem from Computational Biology, using spectral methods to address certain issues with Clark's phasing method.

This project is a mix of expository and exploratory work. Theorems and proofs are, by and large, gleaned from various existing sources such as books and seminal papers, but original results are presented as well. The expository section contains basic definitions, proofs and general observations gleaned from various sources that serve as a gateway to modern ways of thinking about graphs. Missing details of proofs, as presented in the cited sources, have been filled in a way that facilitates the reader's understanding. Original authors are credited wherever possible, though the sources listed do not in any way form an exhaustive list of the wealth of literature available on the subject.

2 Spectral Graph Theory

2.1 Definitions and Theorems: A primer

Let $G = (V, E)$ be an undirected graph, where $|V| = n$. We choose an arbitrary ordering of the vertices, which allows us to abuse notation slightly by letting $V = \{1, \ldots, n\}$. For vertices $i, j \in V$, let $i \sim j$ denote the fact that $\{i, j\} \in E$. Let d_i denote the degree of vertex i. Write $\mathbf{1}_n$, or simply $\mathbf{1}$ when there is no confusion as to what n is, to denote the constant vector $(1, \ldots, 1)$.

Given G, it is natural to define the adjacency matrix $A(G) = (a_{ij})$, where $a_{ij} = 1$ if $i \sim j$, and 0 otherwise. Since G is undirected, A is clearly symmetric. Hence we can appeal to the spectral theorem for symmetric (or more generally, hermitian) matrices from Linear Algebra:

Theorem 2.1. *(Spectral Theorem) Let A be a symmetric $n \times n$ matrix. Then the eigenvalues of A are real, and there exists an orthonormal basis of $\mathbb{R}^n$ consisting of eigenvectors of A.*

The main object of interest is the *combinatorial laplacian matrix* $L(G) = (l_{ij})$, defined as

$$l_{ij} = \begin{cases} d_i & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } i \sim j \\ 0 & \text{else} \end{cases}$$

By definition, we have $L(G) = D(G) - A(G)$, where $D(G)$ is the diagonal matrix whose ith diagonal entry is d_i.

We first note a few properties of L. Note that L is symmetric, so its eigenvalues are real as a consequence of Theorem 2.1. This allows us to list the eigenvalues in nondecreasing order: $\mu_1 \leq \cdots \leq \mu_n$. L's column and row sums are all equal to zero, so we have $L\mathbf{1} = \mathbf{0}$. This tells us that L is singular, and that 0 is an eigenvalue.

There are many ways in which using L to analyze graphs can be useful. One of the earliest applications of the laplacian was noted by Hall [1], in finding a "good" way to embed a connected graph G in $\mathbb{R}^k$. For the 1-dimensional case ($k = 1$), the suggested approach was to minimize the objective function

$$\sum_{\{i,j\} \in E} (x_i - x_j)^2 \tag{1}$$

where we impose the condition that $(x_1, \ldots, x_n) \perp \mathbf{1}$ (since $\mathbf{1}$ is the trivial solution where we place all points at the same location) and $\|x\| = 1$ (since the function is quadratic in x, so we need to avoid scaling by arbitrarily small constants to obtain a better solution).

Why is this a "good" objective function? Spielman [2] provides the following analogy, motivated by a problem in physics. We think of the vertices in the graph as a physical system, where

an edge $\{i, j\}$ represents the connection of i and j by a spring with a spring constant of 1. As a consequence of Hooke's Law, the quadratic form written in Equation 1 quantifies the potential energy stored in the system. It makes sense to try and minimize this function, since such a solution for x would describe the state of the system in equilibrium - all under the provided constraints. [1]

This objective function is related to L, in a very specific way. Observe the following:

$$\begin{aligned} x^T Lx &= \sum_i x_i \left(\sum_j l_{ij} x_j \right) = \sum_{i,j} l_{ij} x_i x_j \\ &= \sum_{\substack{i \neq j \\ i \sim j}} -x_i x_j + \sum_i d_i x_i^2 \\ &= \sum_{\{i,j\} \in E} -2x_i x_j + \sum_i d_i x_i^2 \\ &= \sum_{\{i,j\} \in E} (x_i^2 + x_j^2 - 2x_i x_j) = \sum_{\{i,j\} \in E} (x_i - x_j)^2 \end{aligned}$$

This also shows that L is positive-semidefinite, since the right hand side is a sum of squares. Hence $\mu_1 = 0$ is the smallest eigenvalue of $L(G)$.

The relationship is particularly revealing, especially in light of the following theorem from Linear Algebra:

Theorem 2.2. *(Rayleigh-Ritz) Let A be a real $n \times n$ symmetric matrix. Let $\lambda_1 \leq \cdots \leq \lambda_n$ be its eigenvalues, and let $v_1, \ldots, v_n$ be the corresponding orthonormal basis of eigenvectors. Then*

$$\lambda_1 = \min \frac{x^T Ax}{x^T x}$$

$$\lambda_k = \min_{x \perp \mathit{Span}(v_1, \ldots, v_{k-1})} \frac{x^T Ax}{x^T x}; \quad k = 2, \ldots, n$$

and for each λ_i, the expression on the right hand side is minimized by its respective eigenvector v_i.

Remark. Note that λ_n reduces to $\max \dfrac{x^T Ax}{x^T x}$, since we are minimizing over a 1-dimensional subspace.

By this theorem, we have

$$\min_{\substack{\|x\|=1 \\ x \perp \mathbf{1}}} \sum_{\{i,j\} \in E} (x_i - x_j)^2 = \min_{\substack{\|x\|=1 \\ x \perp \mathbf{1}}} x^T Lx = \min_{x \perp \mathbf{1}} \frac{x^T Lx}{x^T x} = \mu_2$$

[1] This certainly begs the generalization to weighted graphs, where we define $A = (w_{ij})$. A convention is to let $d_i = \sum_j w_{ij}$ and analogously define L as well. The physics argument remains the same, except we set the spring constants to be the edge weights.

and $x = v_2$ is the solution that minimizes Equation 1.

However, this does not tell the whole story. What if we drop the assumption that G is connected? For example, consider the graph obtained by taking the union of two copies of K_3, the complete graph of three vertices. Its laplacian matrix is

$$\begin{pmatrix} -2 & 1 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 1 & 1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & 1 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 & 1 & -2 \end{pmatrix}$$

The kernel of this matrix has dimension 2, and it is spanned by the orthogonal vectors $(0, 0, 0, 1, 1, 1)$ and $(1, 1, 1, 0, 0, 0)$. Using the language of graph embeddings, this tells us that there are actually two trivial orthogonal solutions, where each is given by collapsing the connected components into a separate points on $\mathbb{R}$. This pattern is made precise in the following theorem:

Theorem 2.3. *Let G be an undirected graph. The multiplicity of 0 as an eigenvalue of $L(G)$ is equal to the number of connected components of G.*

Presented below is the standard proof of this theorem, as seen in many sources of literature including [3].

Proof. First, consider the case where G is connected. Let x be a nonzero vector in $\ker L(G)$. Then certainly we have $0 = x^T L x = \sum (x_i - x_j)^2$. Since the right hand side is a sum of squares, each summand must be zero. Therefore, if $i \sim j$, then $x_i = x_j$. Since G is connected, $x_1 = \cdots = x_n$. Hence $\dim \ker L(G) = 1$, which shows that the eigenvalue 0 has multiplicity 1.

To generalize, let $G_1, G_2, \ldots G_m$ be the connected components of G containing $n_1, \ldots, n_m$ vertices respectively. Without loss of generality, let the vertices of G be indexed so that the vertices of G_1 are listed first, then G_2, and so on. Then $L(G)$ is a block-diagonal matrix of the form

$$L(G) = \begin{pmatrix} L(G_1) & & \\ & \ddots & \\ & & L(G_k) \end{pmatrix}$$

It is clear from the above that

$$\ker L(G) = \ker L(G_1) \oplus \cdots \oplus \ker L(G_k)$$

Using the fact that each G_i itself is connected, we conclude

$$\dim \ker L(G) = \sum_{i=1}^{k} \dim \ker L(G_i) = k$$

□

Corollary. *G is connected $\iff$ 0 is a simple eigenvalue of $L(G)$.*

Remark. Due to Theorem 2.3, μ_2 is called the *algebraic connectivity* of G.

It is useful to define the *normalized laplacian* of G, denoted $\mathcal{L}(G)$, as

$$\mathcal{L} = I - D^{-1/2} A D^{-1/2} = D^{-1/2} L D^{-1/2}$$

To avoid technicalities, we define the ith diagonal entry of $D^{-1/2}$ to be zero whenever $d_i = 0$. Observe that $\mathcal{L}$ has the closed form $\mathcal{L} = (\ell_{ij})$, where

$$\ell_{ij} = \begin{cases} 1 & \text{if } i = j \text{ and } d_i \neq 0 \\ \frac{-1}{\sqrt{d_i d_j}} & \text{if } i \neq j \text{ and } i \sim j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

which does not rely on this restriction.

By construction, $\mathcal{L}$ satisfies many properties analogous to those of L:

- $\mathcal{L}$ is singular, and $D^{1/2}\mathbf{1}$ is in the kernel of $\mathcal{L}$.
- $\mathcal{L}$ is positive semidefinite, since for any $x \in \mathbb{R}^n$, $x^T \mathcal{L} x = (D^{-1/2}x)^T L (D^{-1/2}x) \geq 0$.
- Theorem 2.3 is also true: $D^{-1/2}$ is nonsingular, so $\dim \ker \mathcal{L} = \dim \ker L$. Hence the multiplicities of 0 as an eigenvalue are equal.

It also has the following property:

Theorem 2.4. *Let G be connected, and let λ_n be the largest eigenvalue of $\mathcal{L}(G)$. Then $\lambda_n \leq 2$, and equality holds iff G is bipartite.*

Proof. By theorem 2.2,

$$\lambda_n = \max_y \frac{y^T \mathcal{L} y}{y^T y} = \max_x \frac{x^T L x}{x^T D x} = \max_x \frac{\sum_{\{i,j\} \in E} (x_i - x_j)^2}{\sum_i d_i x_i^2}$$

where we have substituted $y = D^{1/2}x$. Observe that

$$\sum_{\{i,j\}\in E}(x_i - x_j)^2 = \sum_{\{i,j\}\in E} 2(x_i^2 + x_j^2) - (x_i + x_j)^2 = 2\sum_i d_i x_i^2 - \sum_{\{i,j\}\in E}(x_i + x_j)^2$$

Plugging it back in, we obtain

$$\lambda_n = 2 - \frac{\sum_{\{i,j\}\in E}(x_i + x_j)^2}{\sum_i d_i x_i^2} \leq 2$$

Assume equality holds for some nontrivial vector x. Since it is a sum of squares, each summand must be zero as well. But this means $i \sim j \implies x_i = -x_j$. Not all of the x_i's are zero, so we can bi-partition the vertices of G by sign. No two positive-valued vertices are connected, and neither are any two negative-valued ones. Therefore, G is bipartite. $\square$

2.2 Normalized Cut of a Graph

Finding graph cuts are a practical application of Spectral Graph Theory. There are many different formulations of graph cuts, but the one we are interested in for this section is the **Normalized Cut**. The definitions here are in the style of Shi & Malik's seminal paper [4] on a spectral method for image segmentation.

Let $G = (V, E)$ be an undirected graph. A *cut* of G is characterized by a non-empty subset $S \subset V$ (or its complement S^c), and it is equal to the number of edges that cross the boundary:

$$cut(S, S^c) = \sum_{u \in S, v \in S^c} a_{uv}$$

where a_{uv} is from the definition of the adjacency matrix, $A(G)$.
We also define the *volume* of a subset $S \subset V$ as

$$vol(S) = \sum_{u \in S} d_u$$

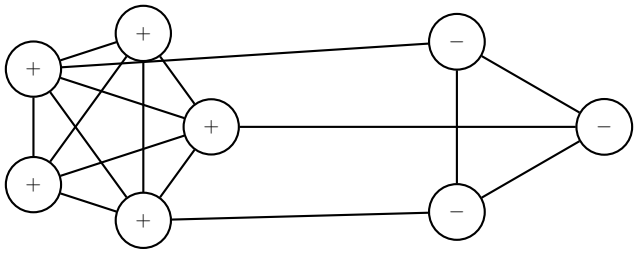
Definition 2.1. The *normalized cut* of G is $\operatorname{argmin}_{S \subset V} NCut(S)$, where $NCut$ is the cost function defined by

$$NCut(S) = cut(S, S^c) \left(\frac{1}{vol(S)} + \frac{1}{vol(S^c)} \right)$$

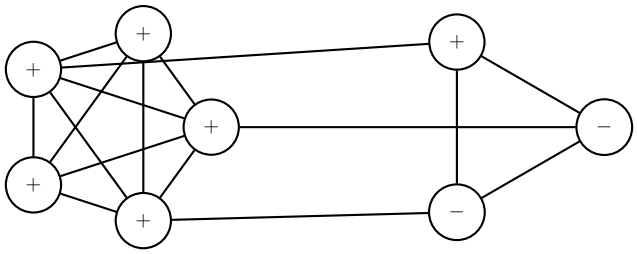
In other words, the cost is equal to the number of cut edges, normalized by the total number of edges connected to each of the partitions. It is reasonable to assume that no vertex is isolated – otherwise there is an obvious optimal cut given by separating that vertex from the rest.

Due to the symmetry and the presence of the second factor in the cost function, the normalized cut tends to be a balanced cut. It also does a pretty good job at preserving clique-like structures in the graph. For example:

Example 2.1. This cut (indicated using plus and minus symbols) has a cost of $3(\frac{1}{23} + \frac{1}{9}) = 0.4638$.



The cut below, however, has a cost of $4(\frac{1}{26} + \frac{1}{6}) = 0.6154$, which is slightly worse.



As such, this type of cut is used often in image segmentation and detections of sub-communities in networks. Computing the normalized cut is known to be NP-hard, due to a proof by Papadimitrou (as noted by [4]). A common approach is to use spectral methods as a heuristic.

We establish a mapping between a partition of V and a family of vectors in $\mathbb{R}^n$.
For $S \subset V$, consider the vector $x = (x_1, \ldots, x_n)$ where

$$x_i = \begin{cases} 1/vol(S) & \text{if } i \in S \\ -1/vol(S^c) & \text{otherwise} \end{cases} \tag{3}$$

Observe that

$$\begin{aligned} x^T L x = \sum_{\{i,j\} \in E} (x_i - x_j)^2 &= \sum_{i \in S, j \in S^c} \left(a_{ij} \left(\frac{1}{vol(S)} + \frac{1}{vol(S^c)} \right)^2 \right) \\ &= \left(\frac{1}{vol(S)} + \frac{1}{vol(S^c)} \right)^2 \cdot cut(S, S^c) \end{aligned}$$

and

$$\begin{aligned} x^T D x = \sum_{i \in E} x_i^2 d_i &= \frac{\sum_{i \in S} d_i}{vol(S)^2} + \frac{\sum_{i \in S^c} d_i}{vol(S^c)^2} \\ &= \frac{1}{vol(S)} + \frac{1}{vol(S^c)} \end{aligned}$$

Therefore,

$$NCut(S, S^c) = \frac{x^T L x}{x^T D x} = \frac{y^T \mathcal{L} y}{y^T y}$$

where $x = D^{-1/2} y$, as before. The Normalized-Cut problem becomes minimizing the above ratio over all cuts S.

To deal with the NP-hardness of this problem, we drop the restriction that x has to be of the form specified by Equation 3, by allowing any vector $y \in \mathbb{R}^n$ that satisfies the normalization property $y^T D^{1/2} \mathbf{1} = 0$. Thus, we obtain the minimization problem

$$\min_{y^T D^{1/2} \mathbf{1} = 0} \frac{y^T \mathcal{L} y}{y^T y}$$

We recall from the previous section that the above is equal to the second smallest eigenvalue λ_2 of $\mathcal{L}$, and it is achieved by letting y be the corresponding eigenvector. The cut S is given by

$$S = \{i \in V : y_i > 0\}$$

Thus, we have the following algorithm:

Algorithm 1 Relaxed Normalized Cut using Graph Laplacian

1: **function** NORMALIZEDCUT($G = (V, E)$)
2: Compute $\mathcal{L} = \mathcal{L}(G)$
3: $x \leftarrow v_2(\mathcal{L})$, the eigenvector corresponding to λ_2
4: $S = \{i : x_i > 0\}$
5: **return** S

Note that the most expensive step is line 3, where we compute $v_2(\mathcal{L})$. Using standard methods, the runtime of computing the eigendecomposition of an $n \times n$ symmetric matrix is $O(n^3)$, and this describes the runtime of Algorithm 1 as well.[2]

[2]Finding faster methods for computing v_2 is an active area of research. More intricate, slightly faster optimizations exist, but are not discussed in detail for simplicity.

2.3 Edge Expansion, Cheeger's Inequality

As we can already see, the algebraic connectivity λ_2 of $\mathcal{L}$ (or μ_2 of L) has a close connection to cuts on the graph. Even when G is not connected as in Theorem 2.1, we have evidence to believe that λ_2 gives us some information about the connectivity of the graph itself.

To investigate this further, we define the Edge Expansion constant, as found in sources including [3], [5], and [6].

Definition 2.2. The *expansion of a cut* $S \subset V(G)$ is defined as

$$\phi(S) = \frac{cut(S, S^c)}{\min(vol(S), vol(S^c))}$$

The *Cheeger constant of* G, also known as the *isoperimetric constant*, is equal to the minimum expansion over all possible nontrivial cuts:

$$\phi(G) = \min_{S \subset V, S \neq \emptyset, S \neq V} \phi(S)$$

We again think of ϕ as a particular cost function of a cut. The relationship between $\phi(G)$ is made precise via Cheeger's Inequality.

Theorem 2.5. *(Cheeger's Inequality for Undirected Graphs) Let* G *be any undirected graph, and let* $0 = \lambda_1 \leq \cdots \leq \lambda_n \leq 2$ *be the eigenvalues of* $\mathcal{L}$*. Then*

$$2\phi(G) \geq \lambda_2 \geq \phi(G)^2/2$$

Note that $\phi(G) = 0$ if and only if G is disconnected. Qualitatively speaking, it is close to zero iff there exists a relatively balanced bipartition, with respect to the volume, with few cut edges. Theorem 2.5 tells us that if λ_2 is very close to 0 (rather than being equal to zero), then the graph is *almost* disconnected - and vice versa. Here, we present a generalization of Trevisan's proof for Cheeger's Inequality on d-regular graphs ([7] [8]).[3] The majority of the work in the proof is in proving Lemma 2.6. First, we need the following definition:

Definition 2.3. Given a graph G on n vertices and a vector $x \in \mathbb{R}^n$, the (volumetric) median of x is the largest number m such that

$$\sum_{x_i < m} d_i \leq \sum_{x_i \geq m} d_i$$

Lemma 2.6. *Let* G *be a graph, and let* $x \in \mathbb{R}^n$ *be a non-constant vector such that* 0 *is the median. Then there exists a cut* S *such that*

$$\phi(S) \leq \sqrt{2 \cdot \frac{x^T L x}{x^T D x}}$$

[3] This proof is an adaptation that summarizes suggestions from these sources that are not made explicit.

Proof. We employ a probabilistic argument to show existence of S. Observe that the quotient $x^T Lx / x^T Dx$ is invariant under scalar multiplication on x, and x cannot be a multiple of $\mathbf{1}$. Thus, we may assume without loss of generality that the vertices are indexed in nondecreasing order of x: $x_1 \leq \cdots \leq x_n$, and x is normalized so that $x_n^2 + x_1^2 = 1$.

Note that $x_1 < 0 < x_n$. Consider the continuous random variable T with density $f(t) = 2|t|$ over the interval $[x_1, x_n]$. Due to our normalization, f does indeed integrate to 1. Let the cut S be defined as

$$S = \{i : x_i \leq T\}$$

For $i \in V$, let D_i be the random variable

$$D_i = \begin{cases} d_i & \text{if the partition that contains } i \text{ has smaller volume} \\ 0 & \text{else} \end{cases}$$

Then we have

$$\mathbb{E}[\min(vol(S), vol(S^c))] = \mathbb{E}\left[\sum_{i=1}^{n} D_i\right] = \sum_{i=1}^{n} \mathbb{E}[D_i] = \sum_{i=1}^{n} d_i \mathbb{P}(D_i = d_i)$$

To compute the right hand side, observe that if $x_i \leq 0$:

$$\mathbb{P}(D_i = d_i) = \mathbb{P}(x_i \leq T \leq 0) = \int_{x_i}^{0} -2t dt = x_i^2$$

Similarly, if $x_i > 0$:

$$\mathbb{P}(D_i = d_i) = \mathbb{P}(0 \leq T < x_i) = \int_{x_i}^{0} 2t dt = x_i^2$$

And so

$$\boxed{\mathbb{E}[\min(vol(S), vol(S^c))] = \sum_{i=1}^{n} d_i x_i^2 = x^T Dx}$$

On the other hand, for each edge e, let C_e be the random variable

$$C_e = \begin{cases} 1 & \text{if } e \text{ is cut by } S \\ 0 & \text{else} \end{cases}$$

So that $\mathbb{E}[cut(S, S^c)] = \sum_{e \in E} \mathbb{E}[C_e] = \sum_{e \in E} \mathbb{P}(C_e = 1)$. If $e = \{i, j\}$ for $i < j$, there are two cases: either x_i, x_j are both the same sign, or they are not. If they are both the same sign, then we can derive the bound

$$\begin{aligned} \mathbb{P}(C_e = 1) = \mathbb{P}(x_i < T < x_j) &= \int_{x_i}^{x_j} 2|t| dt = |x_i^2 - x_j^2| \\ &= |x_i - x_j| \cdot |x_i + x_j| \\ &\leq |x_i - x_j|(|x_i| + |x_j|) \end{aligned}$$

If they are different signs, then instead we have

$$\begin{aligned}\mathbb{P}(C_e = 1) = \mathbb{P}(x_i < T < x_j) &= \int_{x_i}^{x_j} 2|t|dt \\ &= x_i^2 + x_j^2 \\ &\leq |x_i|(|x_i - x_j|) + |x_j|(|x_i - x_j|) \\ &= |x_i - x_j|(|x_i| + |x_j|)\end{aligned}$$

Both bounds are the same expression, so

$$\begin{aligned}\mathbb{E}[cut(S, S^c)] &\leq \sum_{\{i,j\}\in E} |x_i - x_j|(|x_i| + |x_j|) \\ &\leq \sqrt{\sum_{\{i,j\}\in E} (x_i - x_j)^2} \cdot \sqrt{\sum_{\{i,j\}\in E} (|x_i| + |x_j|)^2} \quad \text{(By Cauchy-Schwarz)} \\ &\leq \sqrt{\sum_{\{i,j\}\in E} (x_i - x_j)^2} \cdot \sqrt{\sum_{\{i,j\}\in E} (2x_i^2 + 2x_j^2)} \\ &= \sqrt{\sum_{\{i,j\}\in E} (x_i - x_j)^2} \cdot \sqrt{2\sum_i d_i x_i^2} \\ \implies &\boxed{\mathbb{E}[cut(S, S^c)] = \sqrt{x^T L x} \cdot \sqrt{2x^T D x}}\end{aligned}$$

Putting everything together and using linearity of expectation, we obtain

$$\mathbb{E}\left[cut(S, S^c) - \sqrt{2 \cdot \frac{x^T L x}{x^T D x}} \cdot \min(vol(S), vol(S^c))\right] \leq 0$$

Therefore, there must exist a cut S (induced by some threshold T) so that

$$cut(S, S^c) - \sqrt{2 \cdot \frac{x^T L x}{x^T D x}} \cdot \min(vol(S), vol(S^c)) \leq 0$$

which implies the desired bound. □

Proof. (of Theorem 2.5) The left inequality is the easier of the two to show, since the work was already done in the normalized cut example. For the optimal cut S, consider the vector $(x_1, \ldots, x_n)$, where $x_i = 1/vol(S)$ if $i \in S$, and $-1/vol(S^c)$ otherwise. Again, we have

$$\begin{aligned}\lambda_2 &\leq x^T L x / x^T D x \\ &= cut(S, S^c)\left(\frac{1}{vol(S)} + \frac{1}{vol(S^c)}\right) \\ &\leq \frac{2 \cdot cut(S, S^c)}{\min(vol(S), vol(S^c))} = 2h(G)\end{aligned}$$

Next, we prove the second inequality. Let v be an eigenvector corresponding to λ_2, and make the change of variables $x = D^{-1/2}v$ as we have done previously. We know that $v \perp D^{1/2}\mathbf{1}$, so $x \perp D\mathbf{1}$.

Let m be the median of x. We re-center x, by defining $z = x - m\mathbf{1}$. Observe that z satisfies

$$\begin{aligned}\frac{z^T L z}{z^T D z} &= \frac{\sum_{\{i,j\}\in E}(z_i - z_j)^2}{z^T D z} = \frac{\sum_{\{i,j\}\in E}(x_i - x_j)^2}{(x - x_m\mathbf{1})^T D (x - x_m\mathbf{1})} \\ &= \frac{x^T L x}{x^T D x + (x_m\mathbf{1})^T D (x_m\mathbf{1})} \\ &\leq \frac{x^T L x}{x^T D x} = \lambda_2\end{aligned}$$

z is non-constant and has median 0 by construction, so Lemma 2.6 applies: there exists a cut S such that $\phi(S) \leq \sqrt{2 \cdot \frac{z^T L z}{z^T D z}}$. Hence $\phi(S) \leq \sqrt{2\lambda_2}$, which is equivalent to the desired inequality.

□

2.4 Application: Clique Partitioning

A *clique partition* of a graph G is a partitioning of the vertices $V_1, \ldots, V_k$ such that each subgraph induced by V_i is a clique. Unfortunately, like the other problems in computer science involving cliques, the problem of finding the smallest such partition is NP-complete.

We are also somewhat discouraged from using the multi-way cut proposed by Shi & Malik, a modification that takes first k eigenvectors to perform a cut into k pieces, since we do not know how many pieces we want to begin with.

Here, we suggest a heuristic for obtaining a reasonable solution to MIN-CLIQUE-PARTITION problem in polynomial time, based on Section 2.2. Consider Algorithm 2, which takes advantage of the normalized cut of G:

Algorithm 2 Recursive Clique Partition via Normalized Cuts

1: **function** CLIQUEPARTITION($G = (V, E)$)
2: **if** G is a clique **then return** $\{V\}$
3: $S \leftarrow$ NORMALIZEDCUT(G)
4: $G_1 \leftarrow$ SUBGRAPH(G, S)
5: $G_2 \leftarrow$ SUBGRAPH(G, S^c)
6: **return** CLIQUEPARTITION(G_1) $\cup$ CLIQUEPARTITION(G_2)

Despite the lack of provable approximation bounds, Sections 2.2 and 2.3 give us some intuition about why this is a reasonable thing to try.

Proposition 2.7. *Algorithm 2 has a time complexity of* $O(|V|^4)$.

Proof. This is a rather generous upper bound of the runtime. Let $n = |V|$. To make execution simpler, we will assume that G itself is represented entirely as an adjacency matrix.

Note that line 2 of Algorithm 2 can be done in $O(n^2)$ time, by checking whether $\sum_{i,j} a_{ij}$ equals $\binom{n}{2}$. A similar analysis goes for lines 4 and 5. Line 3 is $O(n^3)$, as we discussed previously. Therefore, the big-O running time of this algorithm is given by the relation $T(n) = T(a) + T(n - a) + n^3$, where $a = |S|$. Typically, the normalized cut tends to be balanced, but we use a liberal

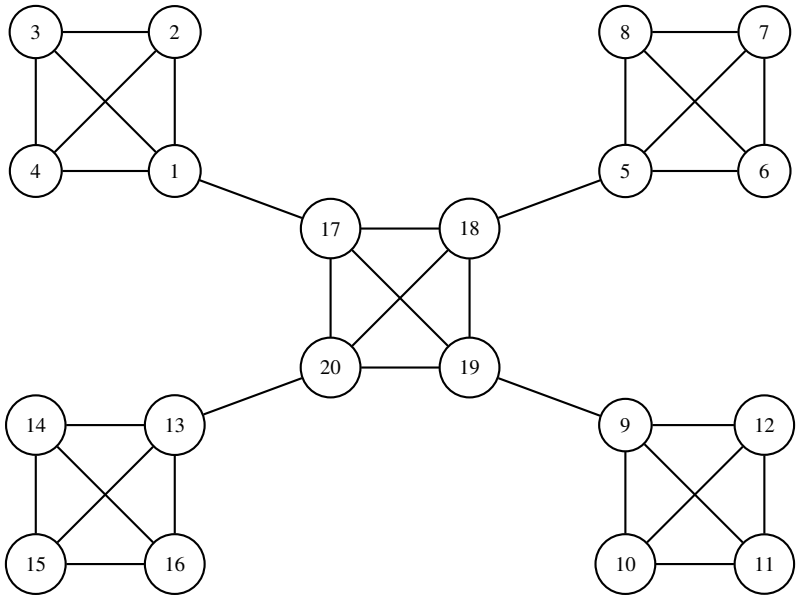
upper bound by setting $a = 1$:

$$\begin{aligned} T(n) &\leq T(1) + T(n-1) + n^3 = c + T(n-1) + n^3 \\ &\leq c + (c + T(n-2) + (n-1)^3) + n^3 \\ &\quad \vdots \\ &\leq nc + \sum_{i=0}^{n} (n-i)^3 \\ &\leq nc + \sum_{i=0}^{n} n^3 = nc + n^4 \end{aligned}$$

□

It is also interesting to consider examples for which the algorithm does not partition cliques correctly. Upon closer inspection, the limitations of our method becomes clear: the *relaxed* normalized cut is, in a way, too balanced! Consider the following example, in which we take a closer look at the first step that produces the suboptimal normalized cut:

Example 2.2. Let G be the graph shown below.



The first few eigenvalues of $\mathcal{L}$ are as follows: $(0.0000, 0.0537, 0.0537, 0.0537, 0.2696, 1.0455, \ldots)$. The fact that λ_2 has multiplicity 3 is not surprising, due to the symmetry of the graph. It suggests that there are three optimal cuts - in the relaxed sense - that are equivalent to each other.

Indeed, the eigenvector v_2 yields the cut $S = \{1, 2, 3, 4, 5, 6, 7, 8, 17, 18\}$. This corresponds to the cut produced by taking the top (or bottom) peripheral copies of K_4 and their conjoined vertices

from the center. This cut could have been made horizontally, vertically, or diagonally, and yields a respectable cut size of 4.

However, the next eigenvalue, $\lambda_5 = 0.2696$, is pretty close to $\lambda_2 = 0.0537$. In fact, v_5 gives us the cut $S = \{17, 18, 19, 20\}$, which is exactly what we wanted. A quick check reveals that the normalized cost using v_2 (or its two symmetrical counterparts) is 0.2353, while the cost of v_5 is 0.1245, so the cut that preserves the rather obvious min-clique partition is indeed better in the normalized sense.

2.5 A Brief Look into Directed Graphs

This section presents the basic premise behind generalizing these ideas to directed graphs. As we will see, very few of the techniques we've employed so far extend naturally, so we have to rely on intuition and hints from the easier, undirected case.

As a generalization, let G be a directed graph, and define $L(G)$'s entries as

$$l_{ij} = \begin{cases} d_i & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

where d_i is the total out-degree of i.

The main difficulty of the directed case comes from the fact that $(i,j) \in E$ does not imply $(j,i) \in E$, so L is no longer symmetric. We can no longer invoke Theorems 2.1 and 2.2, and the eigenvalues can be complex.

Consequently, Theorem 2.3 no longer holds in general. However, we do have the following, weaker result:

Theorem 2.8. *If G is strongly connected, then the eigenvalue 0 is simple.*

Proof. (This is a simplification of the proof by [9], without the language of polynomial rings and field extensions.)

Let $x = (x_1, \ldots, x_n) \in \ker L$ be nonzero. Let $\alpha = \operatorname{argmin}_i x_i$. Since G is strongly connected, we can partition its vertices into the sets $V_0, \ldots, V_d$, where

$$V_k = \{v \in V : \text{Shortest distance from } \alpha \text{ to } v \text{ is } k\}$$

and d is the largest distance from α to v over all $v \in V$.

We will use induction to show that for each k, all vertices $v \in V_k$ satisfies $x_k = x_\alpha$. First, note that $V_0 = \{\alpha\}$, so this is clearly true for $k = 0$. Next, assume that the statement holds for some arbitrary $k \geq 0$. By this hypothesis, observe that for any $i \in V_k$:

$$\begin{aligned} 0 = (Lx)_i &= d_i x_\alpha + \sum_{j \neq i, (i,j) \in E} -x_j \\ &= \sum_{j \neq i, (i,j) \in E} (x_\alpha - x_j) \end{aligned}$$

The right hand side is a sum of strictly nonpositive quantities, since x_α is minimal. Hence each summand is equal to zero, and $(i,j) \in E$ implies $x_j = x_\alpha$. Observe that each vertex in V_{k+1} must have an incoming edge from some vertex in V_k, so $x_j = 0$ for all $j \in V_{k+1}$.

This completes the inductive proof of the claim. We have $x_1 = \cdots = x_n$, so $\dim \ker L = 1$. $\square$

We can also describe the ranges of the spectrum of L, due to Gershgorin.

Theorem 2.9. *(Gershgorin's disk theorem) Let A be a square matrix with entries in $\mathbb{C}$. Then for all eigenvalues λ of A, there exists some i such that*

$$|\lambda - a_{ii}| < \rho_i$$

where $\rho_i = \sum_{j \neq i} |a_{ij}|$.

Proof. (This is the standard proof, as seen in [10].)

Let $(x_1, \ldots, x_n)$ be an eigenvector corresponding to λ. Define $i = \operatorname{argmax}_{i'} |x_{i'}|$. Observe that

$$\begin{aligned} \lambda x_i &= \sum_j a_{ij} x_j \\ \implies (\lambda - a_{ii}) x_i &= \sum_{j \neq i} a_{ij} x_j \\ \implies \lambda - a_{ii} &= \sum_{j \neq i} a_{ij} \frac{x_j}{x_i} \end{aligned}$$

We take the complex norm of both sides, and apply the triangle inequality to obtain:

$$|\lambda - a_{ii}| \leq \sum_{j \neq i} |a_{ij}| \cdot \frac{|x_j|}{|x_i|} \leq \sum_{j \neq i} |a_{ij}| \cdot 1 = \rho_i$$

□

For a graph laplacian L, we have $a_{ii} = -\sum_{j \neq i} a_{ij}$. So as a corollary, we have the following:

Corollary. *Let d be the maximum outdegree of G. All eigenvalues λ of L satisfy*

$$|\lambda - d| < d$$

Furthermore, if G is strongly connected, then all nonzero eigenvalues of L have positive real part.

This gives us some assurance that the eigenvalues are bounded, in a very specific way. Note that the above results are consistent with the undirected case, by thinking of each edge as a pair of directed edges in opposing directions. An interesting problem to consider is to determine whether the converse is true in general (or if not, under which cases), or whether an interesting analogue exists for other notions of connectedness.

The study of directed graph laplacians is largely ongoing research. In very specific cases, useful bounds on λ_2 are known, such as [11] which provides a combinatorial result. In particular, Chung [12] uses the language of random walks and Markov chains to suggest an analogue of $\mathcal{L}$ for the directed case that, unlike L, is symmetrized and has some interesting properties of its own.

3 Haplotype Phasing

3.1 Introduction

It is a well-known fact that the human genome is mostly conserved within a single population. However, there are individual bases in which they may differ, which we refer to as Single Nucleotide Polymorphisms (SNPs). A SNP in a human population is typically biallelic, meaning that there are only two variants of the base from a possibility of four letters in total: A,C,G and T. Humans are diploid organisms, possessing 23 identifiable pairs of chromosomes each.

A *haplotype* refers to the contiguous sequence of alleles for a single, individual chromosomal region, often written as a binary string (e.g. 01001101). The haplotype data is the key to performing analyses on genomic sequences from a population - often to study the evolutionary history of phenotypes or the heritability of diseases. Unfortunately, due to technological limitations, we only have the sequence of allele pairs for a particular chromosomal pair, called a *genotype*. A genotype is represented using a ternary string (e.g. 012201122), where '0' and '1' indicate the two possible homozygous pairs, and a '2' indicates a heterozygous pair. Our goal is to develop a method to infer the haplotype pairs, or *phasings*, for each genotype given a set of input genotypes $\mathcal{G}$.

There are a wealth of references such as [13], [14] and [15] on the different variations of the phasing problem. Many of the formalizations presented here are those used by Prof. Sorin Istrail in CSCI2951-N [16] and in his co-authored paper [17].

Definition 3.1. A haplotype pair h_1, h_2 *explains* a genotype g, or are *complement with respect to* g (denoted $h_1 \oplus h_2 = g$), if

- $h_1[i] = h_2[i] = g[i]$ whenever $g[i] \in \{0, 1\}$, and
- $h_1[i] \neq h_2[i]$ whenever $g[i] = 2$.

Remark. It can be helpful to think of $\oplus$ as a binary, commutative operator on a pair of haplotypes. This is well-defined since $h_1[i]$ and $h_2[i]$ uniquely define $(h_1 \oplus h_2)[i]$, and obviously $h_1 \oplus h_2 = h_2 \oplus h_1$.

Definition 3.2. A haplotype set $\mathcal{H}$ *explains* a genotype set $\mathcal{G}$ if $\forall g \in \mathcal{G}$, $\exists h_1, h_2 \in \mathcal{H}$ such that h_1, h_2 explains g.

The problem of finding the "true" haplotype explanation for a given set of genotypes is not well-defined mathematically. There are different objectives that one can use to measure the quality of a phasing solution.

One is to maximize parsimony - that is, find the simplest possible explanation by finding the smallest $\mathcal{H}$ that explains the input.[4] The Parsimony Haplotype Phasing (PHP) problem was shown to be NP-complete by Hubbell[5], via a reduction from Minimum Clique Partitioning.

Another widely used version is the maximum likelihood phasing problem - which, sadly enough, is also NP-hard. Such a method is iterative, and assumes a weighted distribution of haplotypes based on their frequency. Implemented naively, it is very computationally burdensome based on the length compared to other methods discussed here, but Doug McErlean's undergraduate thesis work at Brown [18] provided a great simplification by exploiting the algebraic symmetries of the likelihood function.

Instead of the above, we will look at a much simpler algorithm that we may be able to improve on using the ideas presented previously in this paper.

3.2 Clark's Method

One of the most elegant algorithms for phasing is Clark's rule-based method [13]. His method tries to exploit the fact that homozygous and single-site heterozygous genotypes have a unique phasing solution. The algorithm iterates, where in each step it tries to phase genotypes using only the known haplotypes from the previous steps.

However, there are a few disadvantages of Clark's method. The first flaw is that the output largely depends on the order of the input. It is a naturally "greedy" method that has some holes in the description, and one must rely on heuristics to complete the algorithm. Second, due to the greedy nature, there is no guarantee of optimality. This method comes with a risk of being left with unresolved genotypes, referred to as "orphans". It is unclear how to proceed if the input has no homozygous or single-site genotypes left at any step, if none of the existing haplotypes explain them.

Can we build a more reliable modification of Clark's method? The heuristic provided in this section suggests a way to phase genotypes without needing trivially solvable genotypes. The idea is to use concepts introduced in Hubbell's proof: we will use the solution for a particular min-clique partition instance to generate a (hopefully) reasonable solution to our instance of the haplotype phasing problem. To this end, we give the following definitions.

Definition 3.3. A haplotype h is *consistent* with a genotype g if $h[i] = g[i]$ whenever $g[i] \in \{0, 1\}$.

Definition 3.4. Two genotypes g_1, g_2 are *consistent* if there exists a haplotype h that is consistent with both.

[4] Any phasing solution has an upper bound of $2|\mathcal{G}|$.

[5] That is, as pointed out by Sharan, Halldorsson, and Istrail in [17]

From this point on, we will assume that the input does not contain any homozygous genotypes, since we are interested in the case where Clark's method fails.

Definition 3.5. Let $\mathcal{G}$ be a set of genotypes. A *(Clark) consistency graph* G of $\mathcal{G}$ is an undirected graph with a bijection $\phi : V(G) \to \mathcal{G}$ in a way such that $\{u, v\} \in E(G) \iff \phi(u)$ is consistent with $\phi(v)$.

Remark. It is clear from the definition that any two such graphs are unique up tto isomorphism, so we will refer to G as "the" consistency graph.

Proposition 3.1. *Let $\mathcal{G}$ be a genotype set such that G is a clique. Then there exists a haplotype h that is consistent with all genotypes in $\mathcal{G}$.*

Proof. Fix any position i, and any arbitrary enumeration $g_1, \ldots, g_n$ of $\mathcal{G}$. The sequence $(g_k[i])_{k=1,\ldots,n}$ cannot contain both a 0 and 1, but may contain anywhere between zero and n 2's. If there is any non-homozygous allele c in the sequence, take $h[i] = c$. Otherwise, take $h[i] = 1$. By construction, $h[i]$ is consistent with the entire sequence. By arbitrarity of i, h is consistent with all of $\mathcal{G}$. □

By this proposition, we can construct a phasing solution for a clique by taking the particular h together with its complementary haplotypes with respect to each genotype. Such a phasing solution always contains at most $n + 1$ haplotypes.

Taking this further: note that if $V_1, \ldots, V_k$ is a clique partition of G, then the resulting phasing solution has at most $\sum_{i=1}^{k}(|V_i| + 1) = |\mathcal{G}| + k$ haplotypes. From this, we apply the following exceedingly simple heuristic, which is far better than resolving orphans at random:

Algorithm 3 Linear-sized phasing solution for orphans

1: **function** PHASE($G = (V, E)$)
2: $V_1, \ldots, V_k = \text{MINCLIQUEPARTITION}(G)$
3: **return** $\bigcup_{i=1}^{k} \text{CLIQUEPHASE}(V_i)$

We are behooved to use some kind of heuristic for line 2. Fortunately, one of our previous results (specifically, Algorithm 2) allows us to find a reasonably-sized clique partition.

3.3 Bounds on Parsimony

There is no guarantee that a phasing solution provided by existing methods will be optimal. In particular, any linear-sized phasing solution will be provably sub-optimal under the parsimony objective, as we will see.

Definition 3.6. Let g_1, g_2, g_3 be distinct genotypes, none of which are completely homozygous. Such a triple is said to be *cyclic* if they are mutually consistent and can collectively be explained using three haplotypes. In other words, they are of the form $g_1 = h_1 \oplus h_2, g_2 = h_1 \oplus h_3, g_3 = h_2 \oplus h_3$ for some distinct haplotypes h_1, h_2, h_3.

Proposition 3.2. *(Characterization of cyclic triples)*
g_1, g_2, g_3 are cyclic $\iff$ there are no positions i such that an odd number of these genotypes are heterozygous at i.

Proof. Assume g_1, g_2, g_3 are cyclic. Then we can write $g_1 = h_1 \oplus h_2$, $g_2 = h_1 \oplus h_3$, $g_3 = h_2 \oplus h_3$. Fix an arbitrary i between 1 and ℓ.

There are only two possibilities: either all three haplotypes are equal at i, or one of them disagrees with the other two. In the first case, $g_1[i] = g_2[i] = g_3[i]$. In the latter, without loss of generality assume $h_1[i] = h_2[i] \neq h_3[i]$. Then $g_1[i] \in \{0, 1\}$ and $g_2[i] = g_3[i] = 2$.

For the reverse direction, we construct an explanation of three haplotypes. For $b \in \{0, 1\}$, let $\overline{b}$ be the other allele. For each i, there are three cases.

- $g_1[i], g_2[i]$ are both homogyzous sites. Then $g_3[i]$ must also be homozygous. They are mutually consistent, go $g_1[i] = g_2[i] = g_3[i]$. Take $h_1[i], h_2[i], h_3[i]$ all equal to $g_1[i]$.
- $g_1[i], g_2[i]$ are both heterozygous sites. Then $g_3[i]$ must be homozygous. Take $h_1[i] = \overline{g_3[i]}$, and $h_2[i] = h_3[i] = g_3[i]$.
- One of $g_1[i], g_2[i]$ are heterozygous - without loss of generality, assume the former. Then $g_3[i]$ must be heterozygous. Take $h_1[i] = h_2[i] = g_1[i]$, and $h_3[i] = \overline{g_1[i]}$.

In all three cases, we have $g_1[i] = h_1[i] \oplus h_2[i]$, $g_2[i] = h_1[i] \oplus h_3[i]$, and $g_3[i] = h_2[i] \oplus h_3[i]$. By arbitrarity of i, we have $g_1 = h_1 \oplus h_2$, $g_2 = h_1 \oplus h_3$ and $g_3 = h_2 \oplus h_3$. □

Proposition 3.3. *Let m be the size of the minimal explanation of $\mathcal{G}$, and let $n = |\mathcal{G}|$. Then $m \in \Omega(\sqrt{n})$.*
In particular, if $\mathcal{G}$ has no cyclic triples and has no purely homozygote genotypes, then $m \geq 2\sqrt{n}$. This bound is tight; there exist examples that achieve equality.

Proof. The first part is a simple combinatorial argument. Let $\mathcal{H}$ denote the minimal explanation, and let $\mathcal{H} \oplus \mathcal{H}$ denote the set $\{h_1 \oplus h_2 : h_1, h_2 \in \mathcal{H}\}$, so $\mathcal{G} \subset \mathcal{H} \oplus \mathcal{H}$. Observe that $|\mathcal{H} \oplus \mathcal{H}| = m^2$. As desired,

$$n \leq m^2 \implies m \geq \sqrt{n}$$

For the second part, define $\mathcal{T} \subset (\mathcal{H} \oplus \mathcal{H}) \setminus \mathcal{H}$ as the largest subset that is free of completely homozygotic genotypes and cyclic triples. By maximality, $|\mathcal{G}| \leq |\mathcal{T}|$, although $\mathcal{G}$ is not necessarily a subset of $\mathcal{T}$. To compute $|\mathcal{T}|$, consider the natural bijection between $(\mathcal{H} \oplus \mathcal{H}) \setminus \mathcal{H}$ and edges of the complete graph K_m, given by $h_i \oplus h_j \mapsto \{v_i, v_j\}$. From this, it is clear that T corresponds to the largest triangle-free subgraph of K_m. By Mantel's Theorem, this maximum is achieved by the maximally balanced, complete bipartite graph $K_{\lfloor m/2 \rfloor, \lceil m/2 \rceil}$, and has at most $\lfloor m^2/4 \rfloor$ elements. Thus, we have

$$n \leq \lfloor m^2/4 \rfloor \leq m^2/4 \implies m \geq 2\sqrt{n}$$

as desired.

This observation also hints at how to construct an example. For an arbitrary $n \geq 1$, let

$$\mathcal{H} = \{h_i : 1 \leq i \leq 2n\}$$

where each haplotype is of the form $h_i = 1 \cdots 101 \cdots 1$, the string of ones that has a zero at the ith position. Consider the bipartition

$$\mathcal{H}_1 = \{h_1, \ldots, h_n\}, \mathcal{H}_2 = \{h_{n+1}, \ldots, h_{2n}\}$$

Observe that if $h_i \oplus h_j = h_{i'} \oplus h_{j'}$ such that $i < j, i' < j'$, then $i = i', j = j'$ since these two genotypes have 2's in the same positions. So if $\mathcal{G} = \mathcal{H}_1 \oplus \mathcal{H}_2$, the set of genotypes defined in the obvious way, then $\mathcal{G}$ has n^2 distinct genotypes.

No two haplotypes have a 0 in the same position, so no genotype has a 0. Hence $\mathcal{G}$ is mutually consistent.

Let $g_1, g_2, g_3 \in \mathcal{G}$, and for the sake of contradiction, assume it is cyclic. Without loss of generality, assume $g_1 = h_1 \oplus h_{n+1}$, so that g_1 is heterozygous at sites 1 and $n+1$. Then one of g_2, g_3 must be of the form $h_1 \oplus h_j$. Then sites $n+1$ and j (greater than $n+1$) both have a single heterozygous site, but by definition no genotype may have two heterozygous sites to the right of n. This is a contradiction, and $\mathcal{G}$ does not contain a cyclic triple.

Clearly, $|\mathcal{G}| = n^2$, and $m = 2n = 2\sqrt{n^2}$. It is also clear that $\mathcal{H}$ is the minimal phasing, since the only alternative would involve explaining some genotype g using the haplotype of all ones: this adds a net total of $+1$ to the explanation size. □

On inputs such as these, even Maximum Likelihood phasing does not always perform well. It often finds the obvious phasing given by taking the haplotype $h = 1 \cdots 1$ of all ones, together with the complements with respect to each genotype. In fact, this is the same solution that we would obtain using the construction provided in Proposition 3.1. [6]

[6] It is likely the case that Parsimony on its own is not the "correct" formulation of the problem. It is important not to forget the biological underpinnings of the problem at some point.

# of genotypes	ML	Clark (random orphans)	Cliques
10	10.9	18.4	12.6
20	13.5	39.8	21.3
30	15.2	46.4	27.6
40	14.8	54.7	28.1
50	15.0	38.6	26.8
60	15.0	65.1	30.3
70	15.0	113.7	47.8
80	15.0	78.0	42.4
90	15.0	101.5	44.5
100	15.0	130.5	39.4

Figure 3.4: This particular simulation was done by starting with an arbitrary, random source of fifteen haplotypes of length 20, and choosing pairs at random to combine into genotypes. To make sure that our algorithm does not devolve into the original method by Clark, we made sure to choose pairs of distinct haplotypes. Each cell represents the average of 30 trials.

3.4 Benchmarks

Despite this, it is still interesting to see how our heuristic fares. Figure 3.4 shows a comparison of Haplotype Explanation sizes for an implementation of an ML-based phaser, versus our modification of Clark's rule-based phaser.

Even prior to taking averages, applying some kind of heuristic to resolve orphans always does better, which is expected since a random phasing always produces a novel, possibly nonsense haplotype.

The advantage to Clark's simple method is speed: The (naive) ML solution always outputs a more parsimonious solution, but takes a long time to execute even for these relatively small inputs, whereas Clark's method finished quickly. Even then, our heuristic doesn't perform all that badly, and the solution size doesn't increase linearly like the randomized scheme does.

References

[1] Kenneth M Hall. An r-dimensional quadratic placement algorithm. *Management science*, 17 (3):219–229, 1970.

[2] Dan Spielman. Spectral graph theory notes (applied math 561). `http://www.cs.yale.edu/homes/spielman/561/`. Iteration: Fall 2014, 2015.

[3] Ravindra B. Bapat. *Graphs and Matrices*. Springer-Verlag London, 1 edition, 2010.

[4] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

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

[6] Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[7] Luca Trevisan. Lecture notes on expansion, sparsest cut, and spectral graph theory. `http://www.eecs.berkeley.edu/~luca/books/expanders.pdf`, .

[8] Luca Trevisan. Spectral graph theory - algorithmic spectral theory boot camp, .

[9] Matthew Thomson and Jeremy Gunawardena. The rational parameterisation theorem for multisite post-translational modification systems. *Journal of theoretical biology*, 261(4):626–636, 2009.

[10] Wikipedia - gershgorin circle theorem. `https://en.wikipedia.org/wiki/Gershgorin_circle_theorem`.

[11] Huang Chao and Ye Xudong. Lower bound for the second smallest eigenvalue of directed rooted graph laplacian. In *Control Conference (CCC), 2012 31st Chinese*, pages 5994–5997. IEEE, 2012.

[12] Fan Chung. Laplacians and the cheeger inequality for directed graphs. *Annals of Combinatorics*, 9(1):1–19, 2005.

[13] Andrew G Clark. Inference of haplotypes from pcr-amplified samples of diploid populations. *Molecular biology and evolution*, 7(2):111–122, 1990.

[14] Laurent Excoffier and Montgomery Slatkin. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Molecular biology and evolution*, 12(5):921–927, 1995.

[15] Dan Gusfield. Haplotype inference by pure parsimony. In *Combinatorial Pattern Matching*, pages 144–155. Springer, 2003.

[16] Sorin Istrail. Csci 2951-n notes. `http://cs.brown.edu/courses/csci2951-n`. Iteration: Fall 2015.

[17] Roded Sharan, Bjarni V Halldórsson, and Sorin Istrail. Islands of tractability for parsimony haplotyping. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 3(3): 303–311, 2006.

[18] Doug McErlean. One constraint to rule them all: How to simplify optimizations under constant variable sum, with applications for maximum likelihood. *Brown University, 2013 CS Honors Thesis*.