CME 305: Discrete Mathematics and Algorithms

Instructor: Professor Aaron Sidford (sidford@stanford.edu)

February 20, 2018

Lecture 12 - Probabilistic Method, Cheeger Inequlity, Random Walks[1]

In this lecture we wrap up our unit on spectral graph theory and begin our discussion of random walks on undirected graphs. First, to further elucidate the proof of Cheeger's inequality we provide some brief coverage of the *probabilistic method*; a powerful tool for showing the existence of combinatorial objects that was central to our proof of Cheeger's inequality. With this established, we provide a brief review of the proof of Cheeger's inequality and discuss some of its extensions and applications. Finally, we conclude the lecture by beginning our discussion of random walks on undirected graphs.

1 Probabilistic Method

We start by a brief discussion of the *probabilistic method*. This is a fundamental proof technique in combinatorial optimization often used to show that certain combinatorial objects exist. The idea that underlies that probabilistic method and what it means to have a proof using the probabilistic method is incredibly simple; however its applications can be quite complex and sophisticated.

The probabilistic method refers to a broad family of strategies for using randomization or randomized processes to provide non-constructive proofs of the existence of objects. One of its most simple instantiation is as follows. Suppose we wish to show that there exists some combinatorial object with some property. To do this, we could simply provide a distribution over combinatorial objects or a probabilistic process for computing them and then try to show that

$$\Pr\left[\text{one of these random objects has the property}\right] > 0\,.$$

If this happens, then clearly there must exist an object with this property. While this is a simply and fairly natural idea, its applications can be quite deep and complex. In the remainder of this section we provide some simple examples of this proof strategy.

Note that we have already seen examples of the probabilistic method, for example, when we showed that there exist at most $\binom{n}{2}$ global minimum cuts of an undirected graph. In the remainder of this section we provide several simple examples of the probabilistic method.

1.1 Coloring Subsets

As our first application of the probabilistic method consider the following problem. Suppose we have a finite set V and a set of subsets $S_1, ..., S_m \subseteq V$ of V. Further suppose that each of these subsets is of size k, i.e. $|S_i| = k$ for all $i \in [m]$. Now we wish to assign one of two colors to each $i \in V$, i.e. either red or blue so that no S_i is monochromatic (i.e. either all red or all blue). Formally, we wish to come up with a partition of V into $R \subseteq V$, the red vertices, and $B \subseteq V$, the blue vertices (i.e. $R \cap B = \emptyset$ and $R \cup B = V$) so that for every $i \in [m]$ it is the case that $S_i \cap R \neq \emptyset$ and $S_i \cap B \neq \emptyset$, i.e. each S_i has at least one red and one blue vertex. Another way of thinking about this problem is that we wish to show there is a subset $C \subseteq V$ that cuts every one of the S_i (i.e. $S_i \cap C \neq \emptyset$ and $S_i \cap (V \setminus C) \neq \emptyset$).

The question we will use the probabilistic method to answer is when can we use m and k to reason that there must exist such a coloring of V so that no S_i is monochromatic. Formally we can show the following.

[1] These lecture notes are a work in progress and there may be typos, awkward language, omitted proof details, etc. Moreover, content from lectures might be missing. These notes are intended to converge to a polished superset of the material covered in class so if you would like anything clarified, please do not hesitate to post to Piazza and ask.

Lemma 1. *Whenever $m < 2^{k-1}$ there exists a coloring so that no S_i is monochromatic.*

Proof. Suppose we randomly color each element of V independently red with probability $1/2$ and blue with probability $1/2$. In other words we pick the red vertices R at random with $\Pr[i \in R] = \frac{1}{2}$ for all $i \in V$ independently at random and let $B = V \setminus R$.

Now under this random coloring, what is the probability any particular S_i is monochromatic? Well, there are 2^k possible assignments of each element $i \in S_i$ to red or blue and each occurs with equal probability. Furthermore, S_i is monochromatic if and only if it is all red or all blue, i.e. if one of two of these 2^k possible assignments happens. Consequently we have that for all $i \in [m]$

$$\Pr[S_i \text{ monochromatic}] = \frac{2}{2^k} = \frac{1}{2^{k-1}}.$$

Furthermore, this implies by union bound that

$$\Pr[\exists \text{ a monochromatic } S_i] = \Pr\left[\cup_{i\in[m]} (S_i \text{ monochromatic})\right] \leq \sum_{i\in[m]} \Pr[S_i \text{ monochromatic}] \leq \frac{m}{2^{k-1}}.$$

Consequently,

$$\Pr[\text{no } S_i \text{ is monochromatic}] = 1 - \Pr[\exists \text{ a monochromatic } S_i] \geq 1 - \frac{m}{2^{k-1}} > 0$$

where in the last step we used that $m < 2^{k-1}$. Consequently, by the probabilistic method we know that there exists a coloring so that no S_i is monochromatic. □

This lemma gives us a bound that allows us to argue that any number of sets that is sufficiently subexponential in k has a monochromatic coloring.

1.2 Clique Subgraphs

A common application of the probabilistic method is to argue about the existence of graphs with certain properties. Here we give a quick application of the lemma in the previous section to argue about the existence of graphs such that neither the graph nor its complement contains a clique. Formally, we say that a undirected unweighted graph $G = (V, E)$ *contains the complete graph on r nodes, K_r,* if it is the case that there is $S \subseteq V$ with $|S| = r$ and $\{i, j\} \in E$ for all $i \neq j \in S$. Furthermore, we define the *complement* of G, denoted $\bar{G} = (V, \bar{E})$ to be the graph where $\{i, j\} \in \bar{E}$ for $i \neq j$ if and only if $\{i, j\} \notin E$. Now the question we wish to ask, is when can we argue that there exists a graph such that K_r is contained neither in it or its complement. Formally, we will show the following.

Lemma 2. *If $\binom{n}{r} < 2^{\binom{r}{2}-1}$ then there exists a n-node graph G such that neither G nor its complement contains K_r.*

Proof. We can prove this lemma by the analysis in the previous section. Let the V of Lemma 1 be the set of the edges of the complete graph on n-nodes and let $S_1, ..., S_m$ for $m = \binom{n}{r}$ denote all possible subsets of edges of the n-node complete graph that induce some clique on r vertices, i.e. for all subsets of r nodes of the complete graph on n-nodes we let the edges between these r-nodes constitute a S_i. Now, note that $|S_i| = \binom{r}{2}$ for all $i \in [m]$ and that $m = \binom{n}{r} < 2^{\binom{r}{2}-1}$. Consequently, by Lemma 1 there is a coloring of V such that no S_i is monochromatic. Note that if we let G be the graph on n vertices where the edges are the blue edges of this coloring than the fact that no S_i is monochromatic directly corresponds to S_i not being a subgraph of either G or its complement. Consequently, neither G nor its complement contains K_r. □

Note that this lemma gives a range of values for n for which neither G or its complement contains K_r.

1.3 Approximate Maximum Cut

As a final brief application of the probabilistic method, lets consider the maximum cut problem where we are given an undirected unweighted graph $G = (V, E)$ and we wish to find a set $S \subseteq V$ to maximize $|\partial(S)|$, i.e. the number of edges cut by S. Whereas we have shown that the minimum s-t cut problem and the global minimum cut problem are solvable with high probability in polynomial time, the same is not known for this maximum cut problem. Moreover, as we will see in the next unit there are reasons to think it might be difficult or impossible to solve the maximum cut problem efficiently in the worst case. However, here we will show that while computing the maximum cut (or even its value) may be difficult it can at least be shown that there is always a cut that cuts a good fraction, in fact 1/2, of the edges of G.

Lemma 3. *Let $G = (V, E)$ be a undirected unweighted graph. There exists $S \subseteq V$ such that $|\partial(S)| \geq \frac{1}{2}|E|$.*

Proof. Let S be the random set generated by putting each $i \in S$ independently with probability 1/2. For such a S and $\{i, j\} \in E$ what is $\Pr[\{i, j\} \in \partial(S)]$, i.e. the probability that edge $\{i, j\}$ is cut by S. Note, regardless of whether $i \in S$ or $i \in V \setminus S$ the probability j is on the same side of the cut S as i is 1/2 and consequently $\Pr[\{i, j\} \in \partial(S)] = 1/2$. Consequently,

$$\mathbb{E}\,|\partial(S)| = \sum_{\{i,j\} \in E} \Pr[\{i, j\} \in \partial(S)] = \frac{|E|}{2}$$

and therefore $\Pr\left[|\partial(S)| \geq \frac{1}{2}|E|\right] > 0$ and by the probabilistic method the result follows. □

2 Cheeger Inequality Recap / Roadmap

With the probabilistic method in mind here we give a brief recap and overview of our proof of Cheeger's inequality. This inequality is quite useful, but its proof is perhaps one of the more complicated ones we will see in this course. Recall the statement of Cheeger's inequality as follows.

Theorem 4 (Cheeger's Inequality)**.** *For simple, undirected, positive weighted, connected G with maximum degree $d_{\max}$ it is the case that*

$$\frac{\sigma(G)^2}{2 \cdot d_{\max}} \leq \lambda_2(\mathcal{L}(G)) \leq 2 \cdot \sigma(G)\,.$$

Where $\lambda_2(\mathcal{L}(G)) = \min_{x \perp \vec{1}} R_{\mathcal{L}}(x)$ for $R_{\mathcal{L}}(x) = \frac{x^\top \mathcal{L} x}{x^\top x}$, $\sigma(G) = \min_{S \subseteq V, S \notin \{\emptyset, V\}} \sigma(S)$ for $\sigma(S) = \frac{w(\partial(S))}{\min\{|S|, |V \setminus S|\}}$, and $d_{\max} = \max_{i \in [n]} \deg(i)$, i.e. the maximum degree of any vertex.

Now to prove Cheeger's inequality, the bound of $\lambda_2(\mathcal{L}) \leq 2 \cdot \sigma(G)$ was fairly straightforward and provable by viewing the optimization of the Rayleigh quotient for λ_2 as a relaxation of the minimization problem in $\sigma(G)$. However, proving the other direction of the inequality that $\frac{\sigma(G)^2}{2 \cdot d_{\max}} \leq \lambda_2(\mathcal{L}(G))$ was more difficult and that is what we will review below.

The main steps in the proof were as follows:

- **Broader Goal**: rather than proving $\frac{\sigma(G)^2}{2 \cdot d_{\max}} \leq \lambda_2(\mathcal{L}(G))$ directly we decided to prove the more general claim that for any $x \perp \vec{1}$ there exists $S \subseteq V$ with $\sigma(S) \leq \sqrt{2 \cdot d_{\max} \cdot R_{\mathcal{L}}(x)}$.
- **Constrain the sets**: to prove this existence we decided to look at a restricted set of sets and show that for $x \perp \vec{1}$ there exists a set $S_c(x) \stackrel{\text{def}}{=} \{i \in V \mid x_i \leq c\}$, known as a *sweep set*, such that $\sigma(S_c(x)) \leq \sqrt{2 \cdot d_{\max} \cdot R_{\mathcal{L}}(x)}$.

- **Simplify the assumptions**: to prove this we showed it sufficed to make some WLOG assumptions on x by scaling and shifting and showed that it sufficed to prove that for $y \in \mathbb{R}^V$ with $y_1 \leq y_2 \leq ... \leq y_n$ with $y_{\lceil \frac{n}{2} \rceil} = 0$ and $y_1^2 + y_n^2 = 1$ then there exists $S_c(y) \stackrel{\text{def}}{=} \{i \in V \mid y_i \leq c\}$, known as a *sweep set*, such that $\sigma(S_c(y)) \leq \sqrt{2 \cdot d_{\max} \cdot R_{\mathcal{L}}(y)}$.
- **Expectation suffices**: to prove this we showed that it sufficed to find a distribution $\mathcal{D}$ over $c \in [y_1, y_n]$ such that
$$\frac{\mathbb{E}_{c \sim \mathcal{D}} w(\partial(S_c(y))}{\mathbb{E}_{c \sim \mathcal{D}} \min\{|S_c(y)|, |V \setminus S_c(y)|\}} \leq \sqrt{2 \cdot d_{\max} \cdot R_L(y)}$$
we showed that this sufficed by a lemma which gave that this implied that
$$\Pr_{c \sim \mathcal{D}} \left[\sigma(S_c(y)) \leq \sqrt{2 \cdot d_{\max} \cdot R_{\mathcal{L}}(y)}\right] > 0$$
and therefore by the **probabilistic method** there exists $S_c(y)$ with $\sigma(S_c(y)) \leq \sqrt{2 \cdot d_{\max} \cdot R_{\mathcal{L}}(y)}$.
- **Analyze a distribution**: finally we completed the proof by showing that for $\mathcal{D}$ with probability density function $p(c) = 2|c|$ it is the case that $\mathbb{E}_{c \sim \mathcal{D}} w(\partial(S_c(y)) \leq \sqrt{2 \cdot d_{\max} \cdot y^\top \mathcal{L} y \cdot y^\top y}$ and $\mathbb{E}_{c \sim \mathcal{D}} \min\{|S_c(y)|, |V \setminus S_c(y)|\} = y^\top y$.

3 Improving the Degree Factor

Recall that Cheeger's inequality shows that $\lambda_2(G)/d_{\max}$ is a multiplicative approximation to $\sigma(G)/d_{\max}$ up to a constant times $\sigma(G)/d_{\max}$. Consequently, when $\sigma(G)/d_{\max}$ is a constant we can use $\lambda_2(G)$ to get a multiplicative constant approximation to $\sigma(G)$, however as $\sigma(G)/d_{\max}$ gets smaller, this approximation worsens. Moreover, since $\sigma(G) \leq \min_{i \in V} \deg(i)$ we see that whenever there is a large ratio between the minimum and the maximum degree in a graph it is the case that Cheeger's inequality can give a very imprecise relation between $\lambda_2(G)$ and $\sigma(G)$.

In certain settings we may wish to have a combinatorial measure of connectivity of a graph that can still be well approximated by eigenvalues even when the the degrees of the graph can vary. There is a variant of Cheeger's inequality that is often used precisely for this purpose. Whereas we saw that Cheeger's inequality involved connecting the Laplacian of a graph to the most well connected graph with uniform degrees we know, i.e. the complete graph, this new variant connects a more *normalized Laplacian* to well connected graphs of non-uniform degrees, known as *product graphs*.

Formally, for a undirected weighted graph $G = (V, E, w)$ with $w \in \mathbb{R}^E_{>0}$ we define the *normalized Laplacian* associated with G by
$$\mathbf{N}(G) \stackrel{\text{def}}{=} \mathbf{D}^{-1/2} \mathcal{L} \mathbf{D}^{-1/2}$$
where $\mathbf{D}$ is the degree matrix with $\mathbf{D}_{ii} = \deg(i)$ for all $i \in V$ and $\mathbf{D}_{ij} = 0$ for $i \neq j$ and $\mathbf{D}^{-1/2}$ being the matrix $\mathbf{D}$ where every diagonal entry is raised to the $-1/2$ power. Note that
$$\mathbf{N}(G) = \mathbf{D}^{-1/2} (\mathbf{D} - \mathbf{A}) \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} .$$
Whereas $\lambda_{\max}(\mathcal{L}) \in [d_{\max}, 2d_{\max}]$ it can be shown that $\lambda_{\max}(\mathbf{N}(G)) \in [1, 2]$ and therefore its eigenvalues are more "*normalized*."

Now, note that $\mathbf{D}$ and $\mathbf{D}^{-1/2}$ are clearly invertible and therefore it is still the case that $\lambda_1(\mathbf{N}(G)) = 0$ and $\lambda_2(\mathbf{N}(G)) > 0$ if and only if G is connected. Consequently, we could still hope for a Cheeger inequality relating $\lambda_2(\mathbf{N}(G))$ to a connectivity measure of G. The measure it is known to be related to is known as the conductance of a graph. Where we define the *conductance of* G to be
$$\phi(G) = \min_{S \subseteq V, S \notin \{\emptyset, V\}} \phi(S)$$

where $\phi(S)$ is the *conductance of a set* S where

$$\phi(S) \stackrel{\text{def}}{=} \frac{w(\partial(S))}{\min\{\text{vol}(S), \text{vol}(V \setminus S)\}}$$

and the volume of a set S, i.e. $\text{vol}(S)$, is defined to be the sum of the degrees of the vertices in S, i.e. $\text{vol}(S) \stackrel{\text{def}}{=} \sum_{i \in S} \deg(i)$. The conductance of a cut is essentially the same as the sparsity of a cut where instead of normalizing by the number of vertices in the smaller side of the cut we normalize by the total degree of the smaller side of the cut.

Note that again the conductance is more normalized than sparsity as $\phi(S) \leq 1$ for all $S \subseteq V$. To see this, note that there is a nice probabilistic interpretation of $\phi(S)$. Let S be a nontrivial subset of the vertices with $\text{vol}(S) \leq \text{vol}(V \setminus S)$. Now suppose we picked a vertex $i \in S$ with probability proportional to $\deg(i)$ and then we pick $j \in V$ with probability proportional to $w_{\{i,j\}}$for $\{i,j\} \in E$. The probability that $\{i,j\} \in \partial(S)$ is precisely $\phi(S)$, i.e. $\phi(S)$ is a natural measure of the probability of a random edge associated with S crossing the cut induced by S.

By modifying the proof we had for Cheeger's inequality the following variant of Cheeger's inequality can be shown for relating $\phi(G)$ to $\lambda_2(\mathbf{N}(G))$.

Theorem 5 (Normalized Cheeger's Inequality)**.** *For simple, undirected, positive weighted, connected G it is the case that*

$$\frac{\phi(G)^2}{2} \leq \lambda_2(\mathbf{N}(G)) \leq 2 \cdot \phi(G)\,.$$

Note that there is no longer a factor of $d_{\max}$ in this equation and thus we see that whenever $\phi(G)$ is a constant then the second smallest eigenvalue of the normalized Laplacian gives a multiplicative constant approximation to the conductance of the graph.

Now, before we saw that $\lambda_2(\mathcal{L}(G))$ and $\sigma(G)$ can be related to how much larger $\mathcal{L}(G)$ is than the complete graph. Similar bounds can be shown for relating $\lambda_2(\mathbf{N}(G))$ and $\phi(G)$ for how much larger $\mathcal{L}(G)$ is than the product graph associated with G. Formally, for $d \in \mathbb{R}^n_{>0}$ we define the product graph induced by d, denoted $K_d = ([n], E_{K_d}, w_{K_d})$ to be the graph where there for all $i \neq j$ there is an edge $\{i,j\}$ with weight $d_i d_j / \|d\|_1$, i.e. the weights of edges are given by the product of entries of d. Note that we have that for all $i \neq j$

$$\mathcal{L}(K_d)_{ij} = \begin{cases} -\frac{d_i d_j}{\|d\|_1} & i \neq j \\ d_i - \frac{d_i^2}{\|d\|_1} & i = j \end{cases}.$$

and therefore

$$\mathcal{L}(K_d) = \mathbf{diag}(d) - \frac{1}{\|d\|_1} d d^\top$$

where $\mathbf{diag}(d)$ is the diagonal matrix with $\mathbf{diag}(d)_{ii} = d_i$. Thus, whereas the Laplacian of the complete graph is a rank one update of the identity matrix, the Lapalcian or a product graph is a rank one update of a diagonal matrix. Similar as to our bounds on $\lambda_2(\mathcal{L}(G))$ and $\sigma(G)$ it can be shown that $\lambda_2(\mathbf{N}(G))$ is the largest value value of α for which $\mathcal{L}(G) \succeq \alpha \cdot \mathcal{L}(K_d)$ where $d \in \mathbb{R}^V$ with $d_i = \deg(i)$ for all $i \in V$ and that up to a constant factor it is the case that $\phi(G)$ is approximately the largest value value of α for which $\mathcal{L}(G) \succeq_{cut} \alpha \cdot \mathcal{L}(K_d)$.

Whether to use the Laplacian or the normalized Laplacian varies from application to application and it is good to know that Cheeger inequalities hold for each.

4 Algorithmic Considerations

So how can we use Cheeger's inequality to find sparse cuts in a graph and therefore cluster it? We have shown that given any $x \perp \vec{1}$ with $R_{\mathcal{L}}(x)$ sufficiently small we can find a sparse cut and therefore it is reasonable to try to simply compute such a vector and use it to cluster, perhaps by using sweep cuts. There are many variants of this algorithm that can be considered and are used in practice. Broadly speaking, this procedure is known as spectral clustering.

Here we briefly discuss how we can use this to get some provable running time bounds for computing provably good approximations to sparsest cut. To do this, we need a procedure for computing approximations to the second smallest eigenvalue and eigenvector of a graph. What we wish to do is compute a vector v that is an ϵ-approximate second smallest eigenvector in the sense that $v \perp \vec{1}$ and $R_{\mathcal{L}}(v) \leq (1-\epsilon)\lambda_2$. If we had this for some constant ϵ then would could use v to find a sweep cut as predicted by Cheeger's inequality up to a constant factor.

So how should we compute an ϵ-approximate eigenvector? One powerful tool for computing approximate eigenvectors in general is known as *power method*. This is commonly stated as a tool for computing the largest eigenvector of of a matrix. Formally, if $\mathbf{M} \in \mathbb{R}^{n \times n}$ is a symmetric positive semidefinite matrix the power method starts with a random vector $x \in \mathbb{R}^n$ with entries as random standard normals and then computes

$$v_k = \frac{\mathbf{M}^k x}{\|\mathbf{M}^k x\|_2}$$

as an approximation to $v_n(\mathbf{M})$. It is not too difficult to show that if $k = \Omega(\epsilon^{-1} \log n)$ then with high probability $R_{\mathbf{M}}(v_k) \geq (1-\epsilon)\lambda_n(\mathbf{M})$. Note that v_k can be easily computed in $O(\text{nnz}(\mathbf{M})k)$ time where $\text{nnz}(\mathbf{M})$ is the number of non-zero entries in $\mathbf{M}$. Consequently, this means that we can compute an ϵ-approximate largest eigenvector of a matrix in nearly linear time for constant ϵ. Furthermore, we can also compute v_k by repeatedly squaring k using fast matrix multiplication and thus v_k can also be computed in $O(n^\omega \log k)$ time, which is faster for very small values of ϵ. Note, even faster running times can also be achieved by using a better polynomial in $\mathbf{M}$, i.e. Chebyshev polynomials, but this is outside the scope of this class.

So how can we use the power method to compute v_2? One idea would be too try to compute the largest eigenvector of $\mathbf{M} = \frac{2d_{\max}}{n}\mathcal{L}(K_n) - \mathcal{L}$. Note that this matrix is PSD and it can be easily shown that the largest eigenvector of $\mathbf{M}$ is the smallest eigenvector of $\mathcal{L}$. Furthermore, we can easily apply $\mathcal{L}(K_n)$ to a vector in $O(n)$ time as $\mathcal{L}(K_n) = n\mathbf{I} - \vec{1}\vec{1}^\top$. Consequently, using the power method we can compute v_k such that

$$\frac{v_k^\top \mathbf{M} v_k}{v_k^\top v_k} \geq (1-\epsilon)\lambda_{\max}(\mathbf{M}) = 2 \cdot d_{\max} - \lambda_2(\mathcal{L})$$

however, since $\vec{1} \in \ker(\mathbf{M})$ we also see that if we add $\vec{1}$ to v_k to make it orthogonal to the all ones vector then this will not change $v_k^\top \mathbf{M} v_k$, however it will decrease $v_k^\top v_k$. Consequently, this gives that in $O((\text{nnz}(\mathcal{L}) + n)\epsilon^{-1} \log n)$ time we can find a vector w_k such that $w_k \perp \vec{1}$ and by scaling $\|w_k\|_2 = 1$ with

$$w_k^\top \mathbf{M} w_k = 2 \cdot d_{\max} - w_k^\top \mathcal{L} w_k \geq (1-\epsilon)\left[2 \cdot d_{\max} - \lambda_2(\mathcal{L})\right]$$

and therefore

$$R_{\mathcal{L}}(w_k) \leq (1-\epsilon)\lambda_2(\mathcal{L}) + \epsilon \cdot d_{\max}\,.$$

Thus, by setting $\epsilon := \epsilon\lambda_2(\mathcal{L})/d_{\max}$ we see that we can compute an ϵ-approximate second smallest eigenvector in $O((\text{nnz}(\mathcal{L})+n)\epsilon(d_{\max}/\lambda_2)\log(n))$ time. In other words, we can compute an ϵ-approximate second smallest eigenvector in nearly linear time whenever $d_{\max}/\lambda_2$ is a constant.

What do we do when λ_2 is not a constant? Here instead we could try to simply run the power method on $\mathcal{L}^\dagger$, the pseudoinverse of the Laplacian, i.e. the matrix that is the inverse outside of its kernel while preserving

the kernel elsewhere or the matrix obtained by inverting the non-zero eigenvalues of the Laplacian and keeping the zero eigenvalues at 0. It is easy to see that the largest eigenvalue of $\mathcal{L}^\dagger$ is the second smallest eigenvector of $\mathcal{L}$ whenever the associated graph is connected. Moreover, an ϵ-approximate top eigenvector of $\mathcal{L}$ for sufficiently small ϵ gives up to a constant an ϵ-approximate v_2. Consequently, if we could solve lienar systems in $\mathcal{L}$ in $O(\mathcal{T})$ time then we could compute $\mathcal{L}^\dagger b$ for any vector b in $O(\mathcal{T})$ time and compute an ϵ-approximate second smallest eigenvector in $O(\mathcal{T}\epsilon^{-1}\log n)$ time with high probability. In fact it was shown in a breakthrough result of Spielman and Teng in 2004 that this is possible where $O(\mathcal{T})$ is nearly linear in the size of the graph for any desired inverse polynomial accuracy in solving a linear system in $\mathcal{L}$ and thus it is possible to compute constant approximations to the second smallest eigenvector with high probability in nearly linear time.

5 More Spectral Graph Theory

Spectral graph theory is an active area of research that has been used recently to achieve faster algorithms for a whole host of combinatorial problems. In fact the nearly linear time Laplacian system solver of Spielman and Teng has been instrumental in turning many spectral graph theory facts into efficient algorithms. We have seen a few connections between the eigenvalues of a Laplacian and combinatorial properties of graphs, but there are many more. For instance there are Cheeger-like inequalities for the largest eigenvalue of a Laplacian, for the smallest few eignevalues of a Laplacian, and more. Moreover, there are extensions of some of these results to directed graphs and hypergraphs. We will see a few more of these connections when we discuss random walks on graphs. If you would like more references, let me know!

6 Random Walks

Here we switch gears and start a new topic about combinatorial properties of graphs. In the remainder of this lecture and next we will discuss random walks on undirected graphs. This is a central topic in graph theory an the study of random processes more broadly. By viewing a graph as specifying a random walk on its vertices many nice properties of the graph emerge. A common tool in data analysis is to view data as graphs and consider these random process as a way of introducing and studying relationships among the vertices of the graph. We cover random walks in this class to give you the tools for further investigation into this area and to highlight connections between random walks on graphs and various concepts we have seen before, e.g. Laplacians of graphs and measures of connectivity.

Without further ado, the central object we will consider in the remainder of this object and next is the standard random walk on an undirected graph. We suppose for the remainder of this section that we have an undirected graph $G = (V, E, w)$ that is simple, connected, and has positive edge weights $w \in \mathbb{R}^E_{>0}$. The standard random walk on this graph is defined as follows. We imagine that we start at some vertex in the graph and then move to a neighboring vertex at random. If we are at a vertex $u \in V$ then the next vertex we go to $v \in V$ is chosen by picking a random edge $\{u, v\} \in E$ with probability proportional to the weight of the edge, i.e. $w_{\{u,v\}}$. A random walk on G is generated by repeating this process, transition to a random neighboring vertex by picking an edge proportional to weight and repeating.

Formally, we call a random sequence of vertices $v_1, ..., v_k \in V$ a random walk on G if independently for all $i \in [k-1$ it is the case that

$$\Pr\left[v_{i+1} = v \,|\, v_i = u\right] = \begin{cases} \frac{w_{\{u,v\}}}{\deg(u)} & \text{if } \{u, v\} \in E \\ 0 & \text{otherwise} \end{cases}.$$

Note that a random walk on undirected graph is an instance of the more general concept of Markov chains, where we undergo a series of probabilistic transitions over a finite set where the probability for each transition depends only on the current state and not the history of states used to get there. This more general theory encompasses random walks on directed graphs as well, which we could have defined similarly, but for simplicity and completeness in this class we will focus on undirected graphs and introduce concepts from Markov chain theory only as needed in this specific case.

6.1 Random Walk Matrix

Before we discuss properties of random walks on graphs here we introduce a natural linear algebraic object associated with the random walk on an undirected graph. Formally, we define the random walk matrix $\mathbf{W}$ associated with G to be $\mathbf{W} = \mathbf{A}\mathbf{D}^{-1}$ where $\mathbf{A}$ is the adjacency matrix of the graph and $\mathbf{D}$ is the diagonal degree matrix associated with the graph. Note that for all $u, v \in V$ we have

$$\mathbf{W}_{vu} = \begin{cases} \frac{w_{\{u,v\}}}{\deg(u)} & \text{if } \{u,v\} \in E \\ 0 & \text{otherwise} \end{cases}$$

and therefore $\mathbf{W}\vec{1}_u$ is the distribution over vertices resulting from one step of a random walk, i.e. $[\mathbf{W}\vec{1}_u]_v$ is the probability we are at vertex v after one step of the random walk from u. Similarly, $\mathbf{W}^2\vec{1}_u$ is the distribution over vertices resulting from two step of a random walk, i.e. $[\mathbf{W}^2\vec{1}_u]_v$ is the probability we are at vertex v after two step of the random walk from u.

6.2 Random Walk Properties

With the basic definitions and notation associated with a random walk established, here we consider some natural questions about random walks we might ask and quantities associated with random walks that we may wish to compute. These quantities correspond to natural properties about graphs we wish to compute or relationships between vertices of the graph that we wish to understand. In the remainder of this lecture and part of next lecture we will discuss how to compute some of these properties efficiently.

- **Stationary Distribution**: a probability distribution over the vertices of the graph, i.e. $s \in \mathbb{R}^V_{\geq 0}$ with $\|s\|_1 = 1$, is a called the stationary distribution of the random walk if $\mathbf{W}s = s$, i.e. one step of the random walk from a vertex sampled by s gives s as a distribution over the vertices.
- **Hitting Time**: the hitting time from a to b, denoted H_{ab}, is the expected number of steps for a random walk starting at a to first reach a vertex b. This is a natural measure of the relationship between two vertices of the graph.
- **Commute Time**: the commute time between a and b, denoted C_{ab}, is the expected number of steps for a random walk starting at a to get to reach b and then go back to a, i.e. $C_{ab} = H_{ab} + H_{ba}$. The commute time is another natural measure of the relationship between two vertices in a graph but unlike hitting time it is symmetric, i.e. while H_{ab} may not equal H_{ba} clearly $C_{ab} = C_{ba}$.
- **Cover Time**: the cover time of a graph is the largest expected number of steps for a random walk to encounter every vertex in the graph given any starting vertex.

6.3 Stationary Distribution

As a warm up, let us determine what the stationary distribution of the random on an undirected graph is. While the stationary distribution of a random walk on directed graphs may be complicated to compute, on

undirected graphs it is quite simple. Note that $\mathbf{W}s = s$ if and only if $(\mathbf{I} - \mathbf{W})s = \vec{0}$. Furthermore, since $\mathbf{W} = \mathbf{A}\mathbf{D}^{-1}$ and since $\mathbf{D}$ is invertible this in turn happens if and only if $(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1}s = 0$. Now we know that $\mathcal{L}x = (\mathbf{D} - \mathbf{A})x = 0$ if and only if $x \in \text{span}(\vec{1})$ when G is connected. and therefore $(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1}s = 0$ if and only if $\mathbf{D}^{-1}s = \alpha\vec{1}$ for some $\alpha \in \mathbb{R}$. Again, since $\mathbf{D}$ is invertible this happens if and only if $s = \alpha d$ where $d \in \mathbb{R}^V$ with $d_i = \deg(i)$ for all $i \in V$. However, since we need $\|s\|_1 = 1$ this means that $\alpha = \frac{1}{\|d\|_1}$ and since $(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1}\left(\frac{1}{\|d\|_1}d\right) = 0$ we have that the $s = \frac{1}{\|d\|_1}d$ is the unique stationary distribution over the vertices in the graph. Consequently, the unique stationary distribution of $\mathbf{W}$ is simply the distribution over vertices of the graph corresponding to picking a vertex with probability proportional to the degree of that vertex.

6.4 Hitting Time

In the next class we will consider the more complicated question of computing hitting times between vertices in a graph. We will show that this can be done efficiently given the ability to solve systems of equations in the Laplacian matrix.