CS 594: Representations in Algorithm Design **Spring 2022**

Lecture (10): Feburary 10, 2022

Lecturer: Xiaorui Sun *Scribe: Jesse Coultas*

1 Last time and today

- Covered concept of an ”Expander” and hitting time.
- Expander Algorithm
- Expander Decomposition

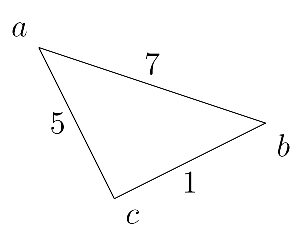
2 Last Time

Last time, the representation of a graph as a Laplasian matrix was presented with introduction of a theorem to bound the expansion of a graph. The edge expansion of a graph is provided as $G(V,E) = \min_{S \subsetneq V} \dfrac{|E(S,\bar{S})|}{\min{(Vol(S), Vol(\bar{S}))}}$. The value $\bar{S}$ represents the complement of S. The hitting time of a random walk is related to the second eigenvalue of the Laplasian matrix as shown by the below theorem.

Theorem 1 (Courant-Fischer and Cheeger’s Inequality)**.**

$$\frac{\lambda_2}{2}(D^{-\frac{1}{2}} \cdot L(G) \cdot D^{-\frac{1}{2}}) \leq \Phi(G) \leq 4\sqrt{\lambda_2}(D^{-\frac{1}{2}} \cdot L(G) \cdot D^{-\frac{1}{2}})$$

The below figure shows a well connected weighted graph. This graph is translated to D, a diagonal matrix of degree. The degree is calculated by summing the weight of edges incident of a. For example, the degree of a is $7 + 5 = 12$ with that value shown at the top/left most value in the matrix. The other vertices are summed in a similar manner on the matrix diagonal.



$$\Rightarrow \mathbf{D} \begin{pmatrix} 12 & & \\ & 8 & \\ & & 6 \end{pmatrix}$$

3 Prove: Courant-Fischer

Note: All eigenvectors are orthogonal

$\tilde{L} = (D^{-1/2} \cdot L(G) \cdot D^{-1/2})$

$\lambda_2 \tilde{L} = \min_{x \perp D} \dfrac{x^T \cdot \tilde{L} \cdot x}{x^T \cdot x}$

Let $y = D \cdot x$

$\lambda_2 \tilde{L} = \min_{y \perp D} \dfrac{y^T \cdot \tilde{L} \cdot y}{y^T \cdot D \cdot y}$

Given $S \subsetneqq V$, $\vec{y_u} = \begin{cases} Vol(\bar{S}), & \text{if } u \in S \\ -Vol(S), & \text{if } u \notin S \end{cases}$

$y = D \cdot \mathbb{1}$

$y^T \cdot L(G) \cdot y \leq 4m^2 \cdot |E(S, \bar{S})|$

$y^T \cdot D \cdot y = 2m \cdot Vol(S)$

$$\frac{y^T \cdot L(G) \cdot y}{y^T \cdot D \cdot y} = 2m \cdot \frac{|E(S,\bar{S})|}{Vol(S) \cdot Vol(\bar{S})} = 2m \cdot \frac{|E(S,\bar{S})|}{\underbrace{\max\{Vol(S), Vol(\bar{S})\} \cdot \min\{Vol(S), Vol(\bar{S})\}}_{\text{Break multiplication of } Vol(S) \text{ and } Vol(\bar{S}) \text{ into a min and max pair.}}}$$

Observe that $m \leq \max\{Vol(S), Vol(\bar{S})\} \leq 2m$, to cancel the m value.

$$\leq 2 \cdot \frac{|E(S,\bar{S})|}{\min\{Vol(S), Vol(\bar{S})\}}$$

4 Expanders

Observation 2. *Expanders are related to the random walk. Given a graph G that is a Φ expander, when G has a small cut size c, then one of the sides of the cut will be smaller than the other. The goal is to cut to cut the graph to have one size much smaller than the other size. The smaller size graph will take less running time. Figure 1 demonstrates a bad cut and figure 2 shows a better cut of the graph.*

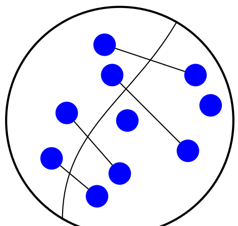


Figure 1: This graph is not a good cut because the difference in volume between the sides is too similar.

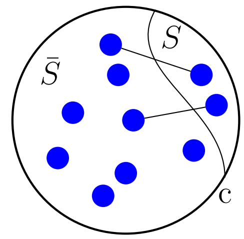


Figure 2: This graph shows a better approach, since the cut results in one side being much smaller than the other.

$$\frac{|E(S,\bar{S})|}{\min\{Vol(S), Vol(\bar{S})\}} \geq \Phi$$

$$c \text{ is equal to } |E(S,\bar{S})| \rightarrow \frac{c}{\min\{Vol(S), Vol(\bar{S})\}} \geq \Phi$$

$$\min\{Vol(S), Vol(\bar{S})\} \leq \frac{c}{\Phi}$$

4.1 Algorithm

Goal: For a given vertex v, find a cut of size c = 1, such that, $V \in S$, $Vol(S) \leq \dfrac{1}{\Phi}$.
Time: $O(\frac{1}{\Phi})^2$

1. Run BFS on v for $\frac{1}{\Phi}$ edges.
2. Let T denote all the edges visited
3. Cut edge $\in T$
4. Check which edges in T is a cut edge
   (a) When found then throw away to see if another cut edge is found
   (b) Enumerate $e \in T$, remove e, run BFS from v for $\frac{1}{\Phi}$ steps.

To generalize in terms of c...

- Time: $O(\frac{c}{\Phi})^{O(c)} \rightarrow O(\frac{c}{\Phi}^4)$
- But need more info from local flow

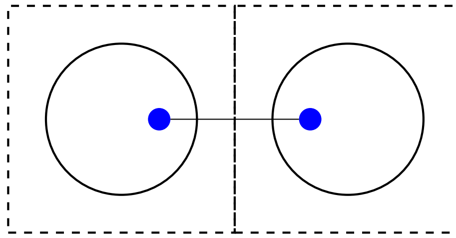


Figure 3: Partition into two cliques, since cliques are good expanders.

- Assume edge is cut, remove and recurse

Lemma 3. *For a Φ expander, the number of cuts size $\leq c$ is $O((n \cdot \frac{c}{\Phi})^c)$*

5 Expander Decomposition

Goal: Partition the graph into pieces that are good expanders.

For graph G with vertex partition P of G, P is a (Φ, ε) expander decomposition if

1. $Q \in P$, $G(Q)$ is a Φ expander.
2. Number of edges crossing expander is at most $\varepsilon \cdot m$.

The value Φ is given related to G. For example, Φ may equal to $\frac{1}{\log n}$.

5.1 Algorithm

Given: G, Φ
Find a (Φ, ε) expander decomposition with εas small as possible for any case.

- If G is Certified as a Φ expander (subset with 1 vertex or $\leq\Phi$), then stop.
- Otherwise...
    1. Graph has a cut $(S, \bar{S})$, such that $\frac{|E(S,\bar{S})|}{\min\{Vol(S), Vol(\bar{S})\}} \leq \Phi$
    2. Find this cut and recurse on $G(S)$ and $G(\bar{S})$

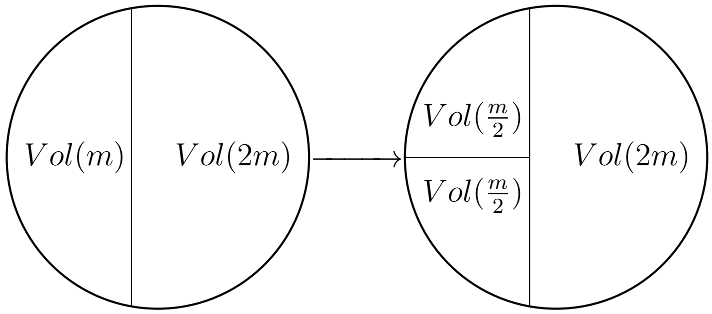


Figure 4: Volume demonstration as algorithm recurses

- Vertex subsets are union of all vertices

Conductance of the graph is expressed by: $\Phi_s(G) = \dfrac{E(S,\bar{S})}{\min\{Vol(S), Vol(\bar{S}\}}$

Finding this cut is NP-Hard. The algorithm does give $(\Phi, \frac{\Phi \log n}{\varepsilon})$ and is the best ε you can get in the worst case. There exists an example such that expander decomposition for Φ expanders has $\geq \Omega(\Phi \cdot \log n \cdot m)$ crossing edges.

5.2 Proof

When encountering a cut, the edges are pooled in set of crossing edges, think of this as a cost $\leq \Phi \cdot \log n \cdot m$. Two key ideas for understanding this algorithm:

1. When finding a cut "charge" the cost to the small side (shown by S in Figure 2). Cost charged to $\dfrac{|E(S,\bar{S})|}{\min\{Vol(S), Vol(\bar{S}\}}$ edges. Given that $Vol(S) \leq Vol(\bar{S})$, each edge (u, v) with $u \in S$ is get charged $\dfrac{|E(S,\bar{S})|}{Vol(S)} \leq \Phi$.

2. Every edge gets charged at most n times with the max charge per edge is $\Phi \cdot \log n$ as shown by Figure 4.