TTIC/CMSC 31150 Mathematical Toolkit **Spring 2013**

Lecture 10: May 6, 2013

Madhur Tulsiani Scribe: Wenjie Luo

In today's lecture, we mainly talked about random walk on graphs and introduce the concept of graph expander, as well as an application of random walk to show its effectiveness.

1 Cheerger's Inequality Recap

Given a d-regular graph with its adjacent matrix A, we define the Laplacian of this graph with adjacent matrix $N = I - \frac{1}{d}A$. Assume A has eigenvalues: $\mu_1, \mu_2, ..., \mu_n$, N has eigenvalues: $\lambda_1, \lambda_2, ..., \lambda_n$, we know:

$$\begin{aligned} d &= \mu_1 \geq \mu_2 \geq ... \geq \mu_n \geq -d \\ 0 &= \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n \leq 2 \end{aligned}$$

Then the Cheeger's Inequality gives that:

$$\frac{\lambda_2}{2} \leq \Phi_G \leq \sqrt{2\lambda_2}$$

where the Φ_G is the expansion of the graph.

2 Random Walk on Graphs

2.1 Basic Idea

First of all, we define the random walk as follows:

- Have a starting vertex.
- At every step, go to a random neighbor of the current vertex.

we use $\mathbf{x} \in \mathcal{R}^n$ to represent the current status, e.g., if we are at i^{th} vertex, than $x_i = 1$ with other items being 0. Then we define the random walk matrix to be M, s.t. $\mathbf{x}^{t+1} = M\mathbf{x}^t$. For a d-regular graph, it's easy to check that $M = \frac{1}{d}A$. We can see that this definition also applies to $\mathbf{x}$ being a distribution. For general graph, we have the random walk matrix to be: $M = D^{-1}A$, where $D_{ii} = deg(i)$, $D_{ij} = 0 \forall i \neq j$. Notice that this matrix has similar eigenvalues to $D^{-1/2}AD^{1/2}$.
Now considering more about d-regular graph, we can see that uniform distribution is a stationary status: $M \cdot \frac{1}{n}\vec{1} = \frac{1}{n}\vec{1}$ since $\vec{1}$ is a eigenvector of M. So the question now is: If we start with any random distribution, how quickly do we converge to the uniform distribution(stationary distribution)?
Explicitly, it progresses as follows:

$$\mathbf{x} \to M\mathbf{x} \to ... \to M^{\perp}\mathbf{x}$$

We split $\mathbf{x}$ to be $\mathbf{x} = \mathbf{x}_\mu + \mathbf{x}_\perp$ where $\mathbf{x}_\mu$ share the direction with stationary distribution and $\mathbf{x}_\perp$ is orthogonal to it. Then we have:

$$\begin{aligned} \mathbf{x}_\mu &= <\mathbf{x}, \frac{1}{\sqrt{n}}\vec{1}> \frac{1}{\sqrt{n}}\vec{1} = \frac{\sum_i x_i}{n} \cdot \vec{1} \\ M\mathbf{x} &= M(\mathbf{x}_\mu + \mathbf{x}_\perp) = \mathbf{x}_\mu + M\mathbf{x}_\perp \end{aligned}$$

Thus, the quantity we care about is $\left\|M^\perp \mathbf{x} - \mathbf{x}_\mu\right\|$. When it's getting close to 0, we are converging to stationary distribution. Consider one step, we have:

$$\|M\mathbf{x} - \mathbf{x}_\mu\| = \|M\mathbf{x}_\perp\| \leq \frac{\mu}{d}\|\mathbf{x}_\perp\| = \frac{\mu}{d}\|\mathbf{x} - \mathbf{x}_\mu\|$$

, where $\mu = \max\{\mu_2, -\mu_n\}$. Then after ℓ steps:

$$\left\|M^\ell \mathbf{x} - \mathbf{x}_\mu\right\| \leq \left(\frac{\mu}{d}\right)^\ell \|\mathbf{x} - \mathbf{x}_\mu\| \leq 2\left(\frac{\mu}{d}\right)^\ell$$

If we set a up-bound: $\left\|M^\ell \mathbf{x} - \mathbf{x}_\mu\right\| \leq \varepsilon$, then

$$2\left(\frac{\mu}{d}\right) \leq \varepsilon \Rightarrow \ell^* = \Omega\left(\frac{\log \varepsilon}{\log \frac{\mu}{d}}\right)$$

So within ℓ^* steps, we can get the distribution very close to stationary distribution.

2.2 Lazy Random Walk

The previous random walk process will go to a new status in each step. Now let's look at another lazy one, where it will stay in the same distribution with 0.5 probability and walk to a new distribution with 0.5 probability. Then the random walk matrix is

$$M' = \frac{1}{2}I + \frac{1}{2}M$$

thus when we have eigenvalues for M:

$$1 = \frac{\mu_1}{d} \geq \frac{\mu_1}{d} \geq ... \geq \frac{\mu_1}{d} \geq -1$$

we have eigenvalues for M' to be:

$$1 = \frac{1}{2} + \frac{\mu_1}{2d} \geq \frac{1}{2} + \frac{\mu_2}{2d} \geq ... \geq \frac{1}{2} + \frac{\mu_n}{2d} \geq 0$$

One convenience of this is that we don't need to consider about both $\mu_2$and μ_n, since this time:

$$\mu' = \max\{\mu_2, -\mu_n\} = \frac{1}{d} + \frac{\mu_2}{2}$$

2.3 Expanders

Now we introduce the concept of "Expander Graph": an expander graph is a sparse graph that has strong connectivity properties, quantified using vertex, edge or spectral expansion. Expander constructions have spawned research in pure and applied mathematics, with several applications to complexity theory, design of robust computer networks, and the theory of error-correcting codes[Wikipedia]. For example, here we use expansion of a graph $\Phi_G \geq 0.1$, then we have:

$$\Phi_G \geq 0.1 \Rightarrow \sqrt{2\lambda_2} \geq \frac{1}{10} \Rightarrow \lambda_2 \geq \frac{1}{200}$$

thus

$$1 - \frac{\mu_2}{d} \geq \frac{1}{200} \Rightarrow \mu_2 \leq \frac{199}{200} d$$

which is equivalent to say $\mu_2 \leq c \cdot d, \quad (c < 1)$.
We hereby give another definition of Cheeger's Inequality:

$$\mu = \max\{\mu_2, -\mu_n\} \leq c \cdot d, \qquad (c < 1)$$

For more information about expander graph and their applications, please refer to this survey[HOORY06].

3 Application of Random Walk

Here we'll use random walk to design an algorithm with less random bits but equivalent performance.

3.1 Problem Setup

First suppose we now have a randomized algorithm which can output whether $\mathbf{x}$ is in $\mathcal{L}$ for any given any $\mathbf{x}$ as:

$$\begin{aligned} &\text{with input:} && \mathbf{x}, \mathbf{r} \\ &\forall \mathbf{x} \in \mathcal{L}, && \mathop{\mathbb{P}}_{\mathbf{r}}[\text{Algo}(\mathbf{x}, \mathbf{r}) = \text{YES}] \geq 1/2 \\ &\forall \mathbf{x} \notin \mathcal{L}, && \mathop{\mathbb{P}}_{\mathbf{r}}[\text{Algo}(\mathbf{x}, \mathbf{r}) = \text{YES}] = 0 \end{aligned}$$

*class of $\mathcal{L}$ for which above algorithm exist is called RP(Randomized Polynomial Time).
Then our objective is to apply expander graph to improve the above inequality of 1/2 to something close to 1.

3.2 Basic Idea

A basic idea for doing that is

- Run the algorithm with ℓ independent $\mathbf{r}_1$, $\mathbf{r}_2$,.. $\mathbf{r}_n$

- Output YES if any run says YES, else output NO.

This algorithm can give us the following conclusion:

$$\forall \mathbf{x} \in \mathcal{L}, \qquad \mathbb{P}_{\mathbf{r}_1,\ldots,\mathbf{r}_n}[\text{Algo}^*(\mathbf{x},\mathbf{r}) = \text{YES}] \geq 1 - \frac{1}{2^\ell}$$
$$\forall \mathbf{x} \notin \mathcal{L}, \qquad \mathbb{P}_{\mathbf{r}_1,\ldots,\mathbf{r}_n}[\text{Algo}^*(\mathbf{x},\mathbf{r}) = \text{YES}] = 0$$

Then if $\|\mathbf{r}\| = R$, Algo* uses $\ell \cdot R$ random bits. Actually we can select those $\mathbf{r}_i$ more wisely so that we can use less random bits to get the same conclusion.

We can achieve the same conclusion with just $O(\ell + R)$ random bits.

3.3 Apply Random Walk on Graph

The algorithm works as follows:
First assume we have an access to an expander graph G with 2^R vertices and $d = O(1)$, for example $d = 10$, thus $\mu = \max\{\mu_2, -\mu_n\} \leq \frac{9d}{10}$. Then we sample $\mathbf{r}_i$ as follows:

- $\mathbf{r}_1$: random vertex of G
- $\mathbf{r}_2$: random neighbor of $\mathbf{r}_1$
- $\vdots$
- $\mathbf{r}_n$: random neighbor of $\mathbf{r}_{n-1}$

Thus the number of random bits we use is: $R + \lceil \log_2 d \rceil \cdot (\ell - 1)$
Now we need to prove that we achieve the same level of accuracy here(also equivalent to say random walk on expander graph is as good as uniform sampling).

Lemma 3.1 *For adjacent matrix A, $A^\ell_{ij} = \#$ walks of length ℓ form $i \to j$*

Proof: First of all, for $\ell = 2$, it's obvious that $A^2_{ij} = \sum_k A_{ik}A_{kj}$
Then by induction, we can find that:

$$A^\ell_{ij} = \sum_{k_1,k_2,\ldots,k_{\ell-1}} A_{ik_1}A_{k_1k_2}\ldots A_{k_{\ell-1}j}$$

with right hand side to be exactly the number of walks of length ℓ ■

Now, let $S \subseteq V$ be a set s.t. $|S| \geq \frac{n}{2}$

Lemma 3.2 $\mathbb{P}\,[$*Random walk of length ℓ never visits S*$] = 2^{-\Omega(\ell)}$

Proof: Given $\mathbf{x} \in \mathcal{L}$, define that:

$$S = \{\mathbf{r} : \text{Algo}^*(\mathbf{x}, \mathbf{r}) = \text{YES}\}$$

From lemma 3.1, we can see the total # walks of length ℓ:

$$(\text{total number})_\ell = \sum_{i,j} A_{ij}^\ell = \vec{1} A^\ell \vec{1} = d^\ell \vec{1}^\top \vec{1} = d^\ell \cdot n$$

Then define a matrix $\overline{A}$ s.t.

$$\overline{A}_{ij} = \begin{cases} 0 & \text{if } i \in S \text{ or } j \in S \\ A_{ij} & \text{otherwise.} \end{cases}$$

then # walks that avoid S is $\vec{1} \cdot \overline{A}^\ell \cdot \vec{1}$. If we can prove that all eigenvalues of $\overline{A}$ are less than d, we are done.
For any $\mathbf{x}$, consider $\mathbf{x}^\top \overline{A} \mathbf{x} = \mathbf{z}^\top A \mathbf{z}$, where

$$z_i = \begin{cases} 0 & \text{if } i \in S \\ x_i & \text{otherwise.} \end{cases}$$

then $\mathbf{x}^\top \overline{A} \mathbf{x} = \sum_{ij} \overline{A}_{ij} x_i x_j = \sum_{ij} A_{ij} x_i x_j = \mathbf{z}^\top A \mathbf{z}$. Similarly as what we did in section 2.1, let $\mathbf{z} = \mathbf{z}_\mu + \mathbf{z}_\perp$, then $\mathbf{z}_\mu = \frac{\sum_i z_i}{n} \cdot \vec{1}$. Now:

$$\begin{aligned} \mathbf{z}^\top A \mathbf{z} &= (\mathbf{z}_\mu + \mathbf{z}_\perp)^\top A (\mathbf{z}_\mu + \mathbf{z}_\perp) \\ &= (\mathbf{z}_\mu + \mathbf{z}_\perp)^\top (d\mathbf{z}_\mu + A\mathbf{z}_\perp) \\ &= \|\mathbf{z}_\mu\|^2 d + < \mathbf{z}_\perp, A\mathbf{z}_\perp > \\ &\leq \|\mathbf{z}_\mu\|^2 d + \mu \|\mathbf{z}_\perp\|^2 \\ &= \|\mathbf{z}_\mu\|^2 d + \mu(\|\mathbf{z}\|^2 - \|\mathbf{z}_\mu\|^2) \end{aligned}$$

while at the same time, we have:

$$\begin{aligned} \|\mathbf{z}_\mu\|^2 &= \sum_i \|\mathbf{z}_m u\|_i^2 = \sum_i (\frac{\sum_i z_i}{n})^2 \\ &\leq \frac{(n - |S|)}{n} (\sum_i z_i)^2 \\ &\leq (1 - \frac{|S|}{n})(\sum_i z_i)^2 \quad \text{recall: } \frac{|S|}{n} \geq \frac{1}{2} \\ &\leq \frac{1}{2} \|\mathbf{z}\|^n \end{aligned}$$

thus, together we get:

$$\begin{aligned} \mathbf{z}^\top A \mathbf{z} &= \|\mathbf{z}_\mu\|^2 d + \mu(\|\mathbf{z}\|^2 - \|\mathbf{z}_\mu\|^2) \\ &\leq (\frac{1}{2} d + \frac{1}{2} \mu) \|\mathbf{z}\|^2 \\ &\leq \frac{d + \mu}{2} \|\mathbf{x}\|^2 \end{aligned}$$

which is to say: $\mathbf{x}^\top \overline{A}\mathbf{x} \leq \frac{d+\mu}{2} \|\mathbf{x}\|^2$. Recall that $\mu = \max\{\mu_2, -\mu_n\} \leq c \cdot d$ $(c < 1)$. Then we have:

$$\mathbb{P}\left[\text{Random walk of length } \ell \text{ never visits } S\right] = \frac{\vec{1}\overline{A}^{\ell}\vec{1}}{\vec{1}A^{\ell}\vec{1}} \leq \frac{\left(\frac{d+\mu}{2}\right)^{\ell} \cdot n}{d^{\ell} \cdot n} = 2^{-\Omega(\ell)}$$

■

Since we know

$$\mathbb{P}\left[\text{Algo}^* \text{ answers NO}\right] = \mathbb{P}\left[\text{Random walk never visits S}\right]$$

thus from lemma 3.2, we can conclude that

$$\forall \mathbf{x} \in \mathcal{L}, \quad \mathbb{P}_{\mathbf{r}_1,\ldots,\mathbf{r}_n}\left[\text{Algo}^*(\mathbf{x},\mathbf{r}) = \text{YES}\right] \geq 1 - 2^{-\Omega(\ell)}$$

i.e. we can achieve the same accuracy as before using only $R + \lceil \log_2 d \rceil \cdot (\ell - 1)$ random bits.

References

[HOORY06] S. HOORY, N. LINIAL and A. WIGDERSON, "Expander Graphs and Their Applications", *Bulletin of the American Mathematical Society*, Oct. 2006, Vol. 43, pp. 439-561.