CAS CS 599 B: Mathematical Methods for TCS

Lecturer: Mark Bun Spring 2022

**Lecture Notes 16:**

**Expanders and Applications**

**Reading.**

- Vadhan Sections 4.1-4.2

# 1 Wrapping Up Random Walks

Given a graph $G$, let $\widetilde{W} = \frac{1}{2}I + \frac{1}{2}AD^{-1}$ be the lazy random walk matrix. Let the eigenvalues of $\widetilde{W}$ be $1 = \omega_1 \geq \omega_2 \geq \cdots \geq \omega_n \geq 0$. Last time, we showed

**Theorem 1.** *For any start vertex $s$, letting $p_0 = \mathbf{1}_{\{s\}}$, and any vertex $v$ and $t \geq 1$, we have*

$$|p_t(v) - \pi(v)| \leq \sqrt{\frac{\deg(v)}{\deg(s)}} \cdot \omega_2^t.$$

Let $d_{\min}, d_{\max}$ be the minimum and maximum degrees of $G$, respectively. Then by summing over all vertices $v$, we get that

$$TV(p_t, \pi) \leq \frac{n}{2}\sqrt{\frac{d_{\max}}{d_{\min}}}\omega_2^t.$$

Thus, to ensure that $p_t$ is $\varepsilon$-close to the stationary distribution in total variation distance, it suffices to take

$$t = O\left(\frac{\log((n/\varepsilon) \cdot (d_{\max}/d_{\min}))}{\log(1/\omega_2)}\right) = O\left(\frac{\log((n/\varepsilon) \cdot (d_{\max}/d_{\min}))}{\nu_2}\right).$$

This quantity is the "mixing time" of the lazy random walk.

Let's look at an example to check if this conforms to our intuition.

**Example 2.** Consider the cycle graph $C_n = \{(i, i+1) \mid i = 1, 2, \ldots, n-1\} \cup \{(n, 1)\}$. Suppose we start a random walk at vertex $n/2$. After $t$ steps, we expect the random walk to reach a vertex $n/2 \pm \Theta(\sqrt{t})$, so the mixing time should probably be something like $t = \Theta(n^2)$.

On Exercise Set 6, you showed that $\nu_2 = O(1/n^2)$. So indeed, the mixing time of the graph is $t = O(n^2 \log n)$.

# 2 Expanders

An expander is a graph with two seemingly contradictory properties: It is $d$-regular for some constant $d$, and in particular sparse, but "looks like" the complete graph in that it is very well connected. Expanders are

ubiquitous in theoretical CS, with applications in network design, graph algorithms, parallel and distributed computing, derandomization, data structures, circuit complexity, coding theory, hardness of approximation, ...

There are several different definitions of expanders. Some are combinatorial, e.g., small sets of vertices are incident to lots of edges, or have large edge boundary. Some are algebraic, e.g., certain eigenvalues of the adjacency matrix or Laplacian are bounded. We'll focus on two definitions and the relationship between them.

**Definition 3** (Edge Expansion)**.** A $d$-regular graph $G$ is a $(k, \varepsilon)$-edge expander if for all sets of vertices $|S| \leq k$, we have $|\partial S| \geq \varepsilon d|S|$.

Better expanders take $k$ to be as large as possible (close to $n/2$) and $\varepsilon$ to be as large as possible (close to 1). Edge expansion is closely related to conductance and isoperimetry: $G$ is an $(n/2, \varepsilon)$-edge expander iff $\phi(G) \geq \varepsilon$.

**Definition 4** (Spectral Expansion)**.** A graph $G$ is a $\gamma$-spectral expander if $\nu_2 \geq \gamma$ and $(2 - \nu_n) \geq \gamma$, where $0 = \nu_1 \leq \nu_2 \leq \cdots \leq \nu_n \leq 2$ are the eigenvalues of the normalized Laplacian $N = \frac{1}{d}L = I - \frac{1}{d}A$.

**Theorem 5** (Equivalence between measures of expansion)**.** *Let $G$ be a $d$-regular graph on $n$ vertices.*

1. *If $G$ is a $\gamma$-spectral expander, then $G$ is an $(n/2, \gamma/2)$-edge expander.*
2. *If $G$ is an $(n/2, \varepsilon)$-edge expander, then $\widetilde{G}$ obtained by adding $d$ self-loops to every vertex in $G$ is an $O(\varepsilon^2)$-spectral expander.*

*Proof.* From the discussion above, the $(n/2, \varepsilon)$-edge expansion of a graph is just the conductance. By Cheeger's inequality,

$$\nu_2/2 \leq \phi(G) \leq \sqrt{2\nu_2}.$$

The LHS tells us that large spectral expansion implies large edge expansion. The other direction follows from the RHS; adding extra self-loops forces $(2 - \nu_n) \geq \nu_2$ for the modified graph. □

# 3 Expander Mixing Lemma

Recall that one interpretation of expanders is that they "look like" the complete graph. One can show that If $G$ is a $\gamma$-spectral expander, then

$$\|N_G - N_K\| \leq 1 - \gamma.$$

Here, $\|\cdot\|$ denotes the spectral norm:

$$\|M\| := \sup_{f \neq 0} \frac{\|Mf\|_2}{\|f\|_2}.$$

**Theorem 6.** *Let $G$ be a $d$-regular, $\gamma$-spectral expander. Then for all subsets $S, T \subseteq V$ with $|S| = \alpha n, |T| = \beta n$, we have*

$$\left| \frac{E(S,T)}{dn} - \alpha\beta \right| \leq (1 - \gamma)\sqrt{\alpha\beta}.$$

*Proof.* Let $N_G = \frac{1}{d}L_G = I - \frac{1}{d}A_G$ be the normalized Laplacian of $G$. Observe that

$$\langle \mathbf{1}_S, N_G \mathbf{1}_T \rangle = \langle \mathbf{1}_S, \mathbf{1}_T \rangle - \frac{1}{d}\langle \mathbf{1}_S, A_G \mathbf{1}_T \rangle = |S \cap T| - \frac{|E(S,T)|}{d}.$$

Similarly if $N_K$ is the complete graph with self-loops,

$$\langle \mathbf{1}_S, N_K \mathbf{1}_T \rangle = |S \cap T| - \frac{|S||T|}{n} = |S \cap T| - \alpha\beta n.$$

Hence, our quantity of interest is

$$\left| \frac{E(S,T)}{dn} - \alpha\beta \right| = \frac{1}{n}\langle \mathbf{1}_S, (N_K - N_G)\mathbf{1}_T \rangle.$$

Now since $\|N_G - N_K\| \leq 1 - \gamma$, we have

$$\begin{aligned} \langle \mathbf{1}_S, (N_K - N_G)\mathbf{1}_T \rangle &\leq \|\mathbf{1}_S\| \cdot \|(N_K - N_G)\mathbf{1}_T\| \\ &\leq \|\mathbf{1}_S\| \|N_K - N_G\| \|\mathbf{1}_T\| \\ &\leq (1-\gamma)\sqrt{\alpha\beta}n. \end{aligned}$$

□

Note that a stronger upper bound of $(1-\gamma)\sqrt{\alpha(1-\alpha)\beta(1-\beta)}$ can be derived by first projecting $\mathbf{1}_S$ and $\mathbf{1}_T$ to be orthogonal to the constant function.

# 4 Random Walks on Expanders

Theorem 1 shows that lazy random walks on expanders mix rapidly. In fact, ordinary walks mix rapidly as well since $\nu_n \ll 2$ implies that expanders are far from bipartite. We can make even more refined statements about the behavior of random walks on expanders. For example:

**Theorem 7.** *If $G$ has spectral expansion $\gamma$ and $v_1, \ldots, v_t$ are a random walk on $G$ with uniformly random start vertex $v_1$, then for all $|B| = \beta n$, we have*

$$\Pr[\{v_1, \ldots, v_t\} \subseteq B] \leq (\beta + (1-\gamma)(1-\beta))^t.$$

Note that this is $\exp(-\Omega(t))$ as long as $\beta, \gamma$ are bounded away from 1.

I might make you prove this as an exercise. Let's see how to use this result to do randomness-efficient error reduction for randomized algorithms. For expanders to be useful for this application, we need to be able to construct them explicitly. There are two flavors of explicitness:

**Weak explicitness** A family $\{G_n\}$ of $d$-regular expanders is *weakly explicit* if for every $n$, the adjacency matrix of $G_n$ can be computed in time $\mathrm{poly}(n)$.

**Strong explicitness** A family is *strongly explicit* if there is an algorithm that given $v \in [n]$ and $j \in [d]$, can compute the $j$'th neighbor of vertex $v$ in $G_n$ in time $\mathrm{poly}(\log n)$.

In many applications, we think of the graph $G_n$ as having size exponential in the input to the problem, which is why we strive for strong explicitness.

### 4.1 Error reduction for coRP

Consider a randomized algorithm $A(x;r)$ for a language $L$ with one-sided error $1/2$. That is, for $x \in \{-1,1\}^n$ and $r \in \{-1,1\}^m$, we have

- For all $x \in L$, we have $\Pr_r[A(x;r)=1] = 1$
- For all $x \notin L$, we have $\Pr_r[A(x;r)=1] \leq 1/2$.

By repeating $A$ roughly $\log(1/\delta)$ times on independent inputs, we can reduce its error on no-instances to $\delta$. This requires $O(m\log(1/\delta))$ random bits. If $A$ runs in time $T(n)$ then the runtime is $O(T(n)\log(1/\delta)$. In Lecture 9, we saw that using, say, pairwise independence would let us reduce the randomness complexity to $O(m+\log(1/\delta))$ at the expense of an increase in runtime to $O(T(n)\cdot(1/\delta))$.

Random walks on expanders let us obtain a different tradeoff:

**Theorem 8.** *Let $\{G\}$ be a strongly explicit family of $d$-regular $\gamma$-spectral expanders with $d = O(1)$ and $\gamma < 1$ a constant. Then there exists a probabilistic algorithm $A'$ deciding $L$ with one-sided error $\delta$ using $m + O(\log(1/\delta))$ random bits and runtime $O(T(n)\log(1/\delta)) + \mathrm{poly}(m)\log(1/\delta)$.*

Let $G$ be a strongly explicit expander on vertex set $V = \{0,1\}^m$. Consider the following algorithm. For a parameter $t$, sample $v_1 \in V$ uniformly at random. For each $i = 1,\ldots,t-1$, pick $v_{i+1}$ to be a random neighbor of $v_i$. Return $\bigwedge_{i=1}^t A(x;v_i)$.

To analyze correctness:

- If $x \in L$, then $\Pr_{v_1,\ldots,v_t}\left[\bigwedge_{i=1}^t A(x;v_i)=1\right] = 1$.
- If $x \notin L$, then letting $B = \{r \mid A(x;r)=1\}$ and noting that $|B| \leq \frac{1}{2}\cdot 2^m$, we have

$$\Pr_{v_1,\ldots,v_t}\left[\bigwedge_{i=1}^t A(x;v_i)=1\right] = \Pr[\{v_1,\ldots,v_t\}\subseteq B] \leq 2^{-\Omega(t)}.$$

Thus, it suffices to take $t = O(\log(1/\delta))$. The calculation of the randomness complexity and runtime bounds follow from strong explicitness.

## 5 Extractors from Expanders

Random walks on expanders also provide a useful method for constructing an extractor. Intuitively, if one starts a random walk from a vertex chosen from a weak random source, and then takes a short random walk, then one ends up at a nearly uniform vertex. We will now make this precise. Recall:

**Definition 9.** A $(k,\varepsilon)$-seeded extractor is a function $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^s \to \{0,1\}^m$ such that $TV(\mathrm{Ext}(X,\mathcal{U}_s),\mathcal{U}_m) \leq \varepsilon$ for every distribution $X$ over $\{-1,1\}^n$ with min-entropy $\geq k$.

Suppose we have a strongly explicit family of $d$-regular $\gamma$-spectral expanders, with $\gamma \geq 1/2$. Consider the following algorithm $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^{t\log d} \to \{0,1\}^n$. On input $x \in \{0,1\}^n$ and $y \in \{0,1\}^{t\log d}$, interpret $y$ as a sequence $y_1,\ldots,y_t \in [d]$ of directions to take. Starting at $x$ in a graph $G$ on vertex set $\{0,1\}^n$, follow these directions and output the destination vertex.

**Theorem 10.** *For $t = (n-k)/2 + \log(1/\varepsilon)$, the algorithm* Ext *described above is a $(k,\varepsilon)$-seeded extractor.*

*Proof.* Let $X$ be a $k$-source, let $p_0$ be the start distribution of the random walk, and let $p_t$ be the distribution of the destination vertex. Letting $U$ denote the uniform distribution on $V$, we have

$$\begin{aligned}
\|p_t - U\|_2^2 &= \|W^t(p_0 - U)\|_2^2 \\
&\leq (1-\gamma)^{2t}\|p_0 - U\|_2^2 \\
&\leq 2^{-2t} \cdot (C(p_0) - 2^{-n}) \\
&\leq 2^{-2t} \cdot (2^{-k} - 2^{-n}) \\
&\leq 2^{-2t-k}.
\end{aligned}$$

Here, we recall that the collision probability $C(p)$ is defined by $\sum_v p(v)^2$.

The first inequality is a consequence of spectral expansion. The second uses the fact that $\|p - u\|^2 = C(p) - 2^{-n}$, which we proved as part of the proof of Lemma 9 in Lecture 12. The third uses the fact that $C(p) \leq 2^{-k}$ for any distribution $p$ with min entropy at least $k$.

Now using these facts in reverse, and setting $t = (n-k)/2 + \log(1/\varepsilon)$ we get

$$C(p_t) = \|p_t - U\|_2^2 + 2^{-n} \leq \frac{1+\varepsilon^2}{2^n}$$

Using Lemma 9 from Lecture 12, which says that $TV(p_t, U) \leq 2^{n/2-1}\sqrt{C(p_t) - 2^{-n}}$, we get that $TV(p_t, U) \leq \varepsilon$. □