12. Explicit construction of constant degree expanders

12.1 Introduction

In this lecture, we will start with the explicit construction of constant-degree expanders. First we will complete the proof of the theorem from the last lecture that random walks on expanders don't stay inside a small set of vertices.

12.2 Random walks on expanders

Theorem 12.1. *Let G be a regular digraph on n vertices, and let B be a set of vertices of cardinality μn. Let $v_1, v_2, \ldots, v_t$ be a random walk on G where v_1 is chosen uniformly from G. Then we have the following.*

$$\Pr\left[\bigwedge_{i=1}^{t} v_i \in B\right] \leq (\mu + \lambda(1-\mu))^t$$

Let P be an $n \times n$ diagonal matrix defined as follows: $P(i,i) = 1$ if $i \in B$, else $P(i,i) = 0$. The probability that a random vertex chosen from G is in B is given by $|uP|_1$ where u is the uniform distribution vector and $|v|_1 = \sum |v_i|$. Conditioned on the first vertex being in B the probability that one step of the random walk stays in B is given by $|uPAP|_1$. Using induction we can prove the following.

Lemma 12.2. *Let G be a regular digraph on n vertices, and let B be a set of vertices of cardinality μn. Let $v_1, v_2, \ldots, v_t$ be a random walk on G where v_1 is chosen uniformly from G. Then we have the following.*

$$\Pr\left[\bigwedge_{i=1}^{t} v_i \in B\right] \leq |uP(AP)^t|_1$$

We saw in the last lecture that the spectral norm of the matrix PAP is bounded.

Claim 12.3. $\|PAP\| \leq \mu + \lambda(1-\mu)$.

Now we give a bound for $\left|uP(AP)^{t-1}\right|_1$. We can write it as

$$\begin{aligned}
\left|uP(AP)^{t-1}\right|_1 &\leq \sqrt{\mu n}\|uP(AP)^{t-1}\| \text{ by Cauchy-Schwarz} \\
&= \sqrt{\mu n}\|uP(PAP)^{t-1}\| \\
&\leq \sqrt{\mu n}\|uP\|\|(PAP)^{t-1}\| \leq \mu(\mu + \lambda(1-\mu))^{t-1} \\
&\leq (\mu + \lambda(1-\mu))^{t-1}.
\end{aligned}$$

12.3 Explicit construction of expanders

Using the previous theorem to perform error reduction is feasible only if we have access to an explicit family of expanders. From hereon, we will refer to an n-vertex, d-regular graph with spectral expansion $1 - \lambda$ as an (n, d, λ)-expander. What do we mean by an explicit family of expanders?

A family $\{G_i\}_i$ of (n, d, λ)-expanders is called a *mildly explicit family* of expanders, if given n, we can compute the graph in the family with n vertices in time poly(n). For using an expander for error-reduction, this level of explicitness is not sufficient since the number of vertices in the graph is exponential. So we require a more stringent notion of explicitness. A family $\{G_i\}_i$ of (n, d, λ)-expanders is called a *fully explicit family* of expanders if given n, v and i, we can compute the i^{th} neighbor of v in the expander on n vertices in time poly$(\log n)$.

Now we will see an iterative method to construct a fully explicit expander family. We will mostly be working with edge-labelled graphs, and we need the following notion of rotation maps.

Rotation maps

Let G be an n-vertex d-regular graph. Supppose that G is an edge labelled graph, i.e. for each vertex v, all the edges incident with v are assigned some label by v. Therefore, for an edge (u, v) it is possible that (u, v) is the i^{th} edge incident on u and the j^{th} edge incident on v. Formally, a rotation map is a representation of the graph that gives these edge labellings.

Definition 12.4 (Rotation maps)**.** *Let G be an n-vertex d-regular graph. The rotation map of G, denoted by* Rot_G *is a function*

$$\mathrm{Rot}_G : [n] \times [d] \to [n] \times [d]$$

such that $\mathrm{Rot}_G(u, i) = (v, j)$ *if the i^{th} edge incident on u is (u, v) and the j^{th} edge incident on v is (u, v).*

For a fully explicit family of expanders, what we want to show is that the rotation map Rot_G is efficiently computable.

Now we need to look at a few operations on graphs and their effects on the expansion of the new graph. These will form the basic blocks of our explicit family of expanders.

Squaring

Let G be an (n, d, λ)-expander. The square of G, denoted by G^2, is a graph on n vertices with degree d^2. The edges of G^2 correspond to paths of length 2 in G. We have already seen that the eigenvalues of the normalized adjacency matrix of G^2 are the squares of the eigenvalues of the normalized adjacency matrix of G. Therefore, we have the following lemma.

Lemma 12.5. *Let G be an (n, d, λ)-expander. Then G^2 is an (n, d^2, λ^2)-expander.*

Notice that the degree of the graph shoots up while squaring the graph. Observe also that given access to Rot_G it is easy to obtain Rot_{G^2}. To get $\mathrm{Rot}_{G^2}(v, (i_1, i_2)$, we need to first get $\mathrm{Rot}_G(v, i_1) = (w, j)$, and then find $\mathrm{Rot}_G(w, i_2)$.

Tensoring

Let G_1 be an (n_1, d_1, λ_1)-expander, and let G_2 be an (n_2, d_2, λ_2)-expander. The tensor product of G_1 and G_2, denoted as $G_1 \otimes G_2$ is defined as follows: $G_1 \otimes G_2$ has $n_1 n_2$ many vertices. Using rotation maps, we can represent $G_1 \otimes G_2$ as follows: $\mathrm{Rot}_{G_1 \otimes G_2}((u_1, u_2), (i_1, i_2)) = (\mathrm{Rot}_{G_1}(u_1, i_1), \mathrm{Rot}_{G_2}(u_2, i_2))$.

A step of a random walk on $G_1 \otimes G_2$ corresponds to one step of the random walk in each G_1 and G_2. If A_1 and A_2 are the respective normalized adjacency matrices of the graphs G_1 and G_2 respectively, then $A_1 \otimes A_2$ is the normalized adjacency matrix of $G_1 \otimes G_2$. The eigevalues of $A_1 \otimes A_2$ is $\{\lambda_i \lambda_j\}$ where λ_is are the eigenvalues of A_1 and λ_js are eigenvalues of A_2. We have the following result about the expansion of the tensor product graph.

Lemma 12.6. *If G_1 is an (n_1, d_1, λ_1)-expander, and G_2 is an (n_2, d_2, λ_2)-expander, then $G_1 \otimes G_2$ is an $(n_1 n_2, d_1 d_2, \max\{\lambda_1, \lambda_2\})$-expander.*

12.3.1 Zig-zag product

In the tensor product of graphs, while we can get a larger graph the degree also shoots up. To get an explicit expander family, we need a sequence of graphs with larger and larger number of vertices and constant degree. To that end, we come to the concept of zig-zag product of graphs.

Let G_1 be an (n_1, d_1, λ_1)-expander, and let G_2 be a (d_1, d_2, λ_2)-expander. The zig-zag product of G_1 and G_2, denoted by $G_1 ⓩ G_2$ is an $(n_1 d_1, d_2^2, f(\lambda_1, \lambda_2))$-expander defined in the following way. Let $(i_i, i_2) \in [d_2] \times [d_2]$, and let $v[i]$ denote the i^{th} neighbor of a vertex v. For a vertex $(u, j) \in G_1 ⓩ G_2$, we obtain the $(i_1, i_2)^{th}$ neighbor as follows: Let $j' = j[i_1]$, and let $v = u[j']$, and let $j'' = j'[i_2]$. Then $(u, j)[i_1, i_2] = (u, j'')$.

We can think of the graph $G_1 ⓩ G_2$ as graph on n_1 "clouds", where each cloud contains the graph G_2. A random walk on $G_1 ⓩ G_2$ corresponds to taking a random step in the cloud(zig), then moving to a different cloud based on the vertex within the first cloud, and then taking a random step in the new cloud(zag). We will analyse this random walk in more detail in the coming lectures.