Expander Graphs and Their Applications (IX)

Yijia Chen
Shanghai Jiaotong University

Review of the Previous Lecture

The Base Graphs

Recall:

1. In the inductive construction of expander graph family using the zig-zag product, we start with a $(d^4, d, 1/4)$-graph.

2. In Reingold's algorithm, we start with a $(d^{16}, d, 1/2)$-graph.

In the following, we

- provide some *explicit construction*,
- and prove their existence by the *probabilistic method*.

The Affine Plane

Let $q := p^t$ where p is a prime and $t \in \mathbb{N}$. And let $\mathbb{F}_q$ be the *finite field of size* q.

Then AP_q is a graph with vertex set $\mathbb{F}_q^2$ and edge set

$$\big\{\text{an edge between } (a,b) \text{ and } (c,d) \\ \mid a,b,c,d \in \mathbb{F}_q \text{ and } ac = b + d\big\}.$$

Equivalently, we connect the vertex (a,b) to all points on the line

$$L_{a,b} := \big\{(x,y) \mid y = ax - b\big\}.$$

Lemma

AP_q is a $(q^2, q, 1/\sqrt{q})$-graph.

The Affine Plane (cont'd)

Let

$$AP_q^1 := AP_q \otimes AP_q$$
$$AP_q^{i+1} := AP_q^i \circled{z} AP_q.$$

Theorem
AP_q^i is an $(q^{2(i+1)}, q^2, i/\sqrt{q})$-graph.

Choosing some sufficiently large q, we can get a $(d^4, d, 1/4)$-graph or a $(d^{16}, d, 1/2)$ graph.

Main Theorem

Theorem

There exists a constant $c > 0$ such that for all sufficiently large $n \in \mathbb{N}$ three exists an n-vertex, 3-regular graphs with $h(G) \geq c$.

Random Perfect Matching

Definition

Let G be a graph. A matching M of G is a subset of $E(G)$ (without selfloop) such that *every vertex appears in at most one edge of the subset.* M is a perfect matching of G if every vertex is incident to one edge in M.

Let $k \in \mathbb{N}$ and $V := [2k]$. Consider the following random process $\mathbb{P}(k)$:

1. $S \leftarrow V$ and $E \leftarrow \emptyset$.
2. **while** $S \neq \emptyset$ **do**
3. Choose a pair $(u, v) \in S^2$ uniformly at random.
4. $S \leftarrow S \setminus \{u, v\}$ and $E \leftarrow E \cup \{$an edge between u and $v\}$.
5. Output E.

$\mathbb{P}(k)$ is a random perfect matching on $[2k]$.

Random d-Regular Graph

Let $k, d \in \mathbb{N}$ and consider the random process $\mathbb{R}_d(k)$.

1. $V \leftarrow [2k]$ and $E \leftarrow \emptyset$.
2. **for** $\ell = 1$ **to** d **do**
3. $\quad E \leftarrow E \cup \mathbb{P}(k)$
4. Output (V, E).

$\mathbb{R}_d(k)$ is a d-regular graph on vertices $[2k]$.

An Important Warning: $\mathbb{R}_d(k)$ is not uniformly distributed over all d-regular graphs on vertices $[2k]$.

Main Theorem (Restated)

Theorem
There exists a constant $c > 0$ such that for all sufficiently large $k \in \mathbb{N}$

$$\Pr\left[h(\mathbb{R}_3(k)) \geq c\right] > 0$$

Introduction to PCP

Probabilistic Verifier

Definition

A verifier V is a *probabilistic polynomial time algorithm* with access to an input $x \in \Sigma^*$ and a string $\tau \in \Sigma^*$ of (internal) *random binary bits*. Furthermore, V has access to a proof $\pi \in \Sigma^*$.

V will either accept or reject the input x, depending on (x, τ, π).

We require that V is non-adaptive, i.e., it first reads the input x and the random bits τ, and then decides which positions in the proof τ it wants to query. That is, the positions V queries do not depend on the answers that V got from *previous queries*.

The result of V's computation on x, τ and π is denoted by $V(x, \tau, \pi)$. As usual 1 means *accepting*, 0 for *rejecting*.

Probabilistic Verifier (cont'd)

1. V is polynomial time in $|x|$.
2. The query positions depend on and only on x and τ (non-adaptive), so we might view the query positions as a function

$$p_V : \Sigma^* \times \Sigma^* \to \mathbb{N}^*,$$

   such that $p_V(x, \tau)$ is computable in time polynomial in $|x|$. We may assume p_V is given *explicitly* with the verifier V.
3. The run of $V(x, \tau, \pi)$ that does not involve π might also depend on the random string τ.
4. We can divide the computation of $V(x, \tau, \pi)$ into 3 parts:
   (i) Compute from x and τ a sequence of positions $\bar{p} = p_1 p_2 \ldots p_\ell = p_V(x, \tau)$.
   (ii) Read the letters on the positions $\bar{p}$ of π to form a string

$$\pi \restriction \bar{p} := \pi(p_1)\pi(p_2)\ldots\pi(p_\ell),$$

   where $\underline{\pi(p)}$ denotes the letter in the p-th position of π.
   (iii) Compute an answer from x, τ, and $\pi \restriction \bar{p}$.

$(r(n), q(n))$-Restricted Verifier

Definition

Let $r, q : \mathbb{N} \rightarrow \mathbb{N}$ be two monotone functions.

An $(r(n), q(n))$-restricted verifier is a verifier that for inputs of length n uses at most $O(r(n))$ random bits and queries at most $O(q(n))$ bits from the proof.

1. If V is $(r(n), q(n))$-restricted, then for any input $x \in \Sigma^*$, Then $|\tau| = O(r(n))$ and $|p_V(x, \tau)| = O(q(n))$.
2. There is no restriction on the length of the proof π.

PCP Classes

Definition

Let $r, q : \mathbb{N} \to \mathbb{N}$ be two monotone functions.
The class $\text{PCP}(r(n), q(n))$ consists of all languages Q where there exists an $(r(n), q(n))$-restricted verifier V such that for all x

$$x \in L \iff \exists \pi \ \Pr_{\tau}[V(x, \tau, \pi) = 1] = 1,$$

$$x \notin L \iff \forall \pi \ \Pr_{\tau}[V(x, \tau, \pi) = 1] < \frac{1}{2}.$$

Here in $\Pr_{\tau}[\dots]$ the probability is taken over all random strings τ of length $O(r(n))$.

Remark. $1/2$ *in the above definition can be replaced by any* $0 < \varepsilon < 1$.

Definition

Let $R, Q \subseteq \mathbb{N} \to \mathbb{N}$ be two classes of monotone functions.

$$\text{PCP}(R, Q) := \bigcup_{r \in R, q \in Q} \text{PCP}(r(n), q(n)).$$

Let poly denote the class of *all polynomials over* $\mathbb{N}$, i.e. $\mathbb{N}[x]$.

Theorem

$$\begin{aligned} \mathrm{PCP}(\texttt{poly}, 0) &= \mathrm{coRP}, \\ \mathrm{PCP}(0, \texttt{poly}) &= \mathrm{NP}. \end{aligned}$$

Proof of $\mathrm{PCP}(\mathtt{poly}, 0) = \mathrm{coRP}$

For every problem $Q \in \Sigma^*$, its complement is

$$\bar{Q} := \{x \in \Sigma^* \mid x \notin Q\}.$$

Let $Q \subseteq \Sigma^*$ with $\bar{Q} \in \mathrm{RP}$. Then there is a polynomial time probabilistic algorithm $\mathbb{A}$ such that for every $x \in \Sigma^*$

$$\begin{aligned} x \notin Q &\iff x \in \bar{Q} \iff \Pr_\tau[\mathbb{A}(x,\tau) = 1] \geq \tfrac{1}{2}, \\ x \in Q &\iff x \notin \bar{Q} \iff \Pr_\tau[\mathbb{A}(x,\tau) = 1] = 0, \end{aligned}$$

where τ is a random string of length *polynomial in $|x|$*.
By *swapping accepting and rejecting* of $\mathbb{A}$ we get an algorithm $\bar{\mathbb{A}}$:

$$\begin{aligned} x \notin Q &\iff \Pr_\tau[\bar{\mathbb{A}}(x,\tau) = 0] \leq \frac{1}{2}, \\ x \in Q &\iff \Pr_\tau[\bar{\mathbb{A}}(x,\tau) = 0] = 1. \end{aligned}$$

That is, $Q \in \mathrm{PCP}(\mathtt{poly}, 0)$.

$\mathrm{PCP}(\mathtt{poly}, 0) \subseteq \mathrm{coRP}$ can be proved similarly. $\dashv$

Proof of $\mathrm{PCP}(0, \mathtt{poly}) = \mathrm{NP}$

$\mathrm{NP} \subseteq \mathrm{PCP}(0, \mathtt{poly})$ is trivial.

Let $Q \in \mathrm{PCP}(0, \mathtt{poly})$. Since no random bit is needed, there is a polynomial time deterministic verifier V such that

$$x \in Q \iff \exists \pi \ V(x, \pi) = 1,$$
$$x \notin Q \iff \forall \pi \ V(x, \pi) = 0.$$

with $|p_V(x)| = q(|x|)$ for some polynomial q.

Consider the following algorithm $\mathbb{A}(x)$.

1. $\ell \leftarrow q(|x|)$.
2. *Guess a string* $\bar{a} = a_1 \dots a_\ell \in \Sigma^\ell$.
3. $\bar{p} = p_1 \dots p_\ell \leftarrow p_V(x)$.
4. Simulate $V(x, \pi)$ for an "imaginary" π by replacing each π_{p_i} by a_i for $1 \leq i \leq \ell$.

If $x \in Q$, then we have a π witnessing the membership. Thus $\mathbb{A}$ can guess $\bar{a} = \pi \restriction p_V(x)$, and accept x.
If $x \notin Q$, then there is no way for $\mathbb{A}$ to guess an $\bar{a}$ to accept x. $\dashv$