Chapter 23

The Probabilistic Method II

By Sariel Har-Peled, March 19, 2024[1]

> “Today I know that everything watches, that nothing goes unseen, and that even wallpaper has a better memory than ours. It isn’t God in His heaven that sees all. A kitchen chair, a coat-hanger a half-filled ash tray, or the wood replica of a woman name Niobe, can perfectly well serve as an unforgetting witness to every one of our acts.”
>
> Gunter Grass, The tin drum

23.1. Expanding Graphs

In this lecture, we are going to discuss *expanding graphs*.

Definition 23.1.1. An (n, d, α, c) ***OR-concentrator*** is a bipartite multigraph $\mathsf{G}(L, R, E)$, with the independent sets of vertices L and R each of cardinality n, such that

(i) Every vertex in L has degree at most d.

(ii) Any subset S of vertices of L, with $|S| \leq \alpha n$ has at least $c\,|S|$ neighbors in R.

A good (n, d, α, c) OR-concentrator should have d as small as possible[2], and c as large as possible.

Theorem 23.1.2. *There is an integer n_0, such that for all $n \geq n_0$, there is an $(n, 18, 1/3, 2)$ OR-concentrator.*

Proof: Let every vertex of L choose neighbors by sampling (with replacement) d vertices independently and uniformly from R. We discard multiple parallel edges in the resulting graph.

Let $\mathcal{E}_s$ be the event that a subset of s vertices of L has fewer than cs neighbors in R. Clearly,

$$\mathbb{P}[\mathcal{E}_s] \leq \binom{n}{s}\binom{n}{cs}\left(\frac{cs}{n}\right)^{ds} \leq \left(\frac{ne}{s}\right)^{s}\left(\frac{ne}{cs}\right)^{cs}\left(\frac{cs}{n}\right)^{ds} = \left[\left(\frac{s}{n}\right)^{d-c-1}\exp(1+c)\,c^{d-c}\right]^{s},$$

since $\binom{n}{k} \leq \left(\frac{ne}{k}\right)^{k}$. Setting $\alpha = 1/3$ using $s \leq \alpha n$, and $c = 2$, we have

$$\begin{aligned}\mathbb{P}[\mathcal{E}_s] &\leq \left[\left(\frac{1}{3}\right)^{d-c-1} e^{1+c}c^{d-c}\right]^{s} \leq \left[\left(\frac{1}{3}\right)^{d} 3^{1+c}e^{1+c}c^{d-c}\right]^{s} \leq \left[\left(\frac{1}{3}\right)^{d} 3^{1+c}e^{1+c}c^{d}\right]^{s}\\ &\leq \left[\left(\frac{c}{3}\right)^{d}(3e)^{1+c}\right]^{s} \leq \left[\left(\frac{2}{3}\right)^{18}(3e)^{1+2}\right]^{s} \leq (0.4)^{s},\end{aligned}$$

as $c = 2$ and $d = 18$. Thus,

$$\sum_{s\geq 1}\mathbb{P}[\mathcal{E}_s] \leq \sum_{s\geq 1}(0.4)^{s} < 1.$$

It thus follows that the random graph we generated has the required properties with positive probability. ■

[1]

[2] Or smaller!

23.1.1. An alternative construction

Theorem 23.1.3. *Consider a bipartite graph over left and right sets L and R, such that $n = |L| = |R|$. Consider a random graph* G *formed by the union of $d = 18$ random perfect matchings between L and R. Let* G *be the resulting graph. Then, for $d \geq 18$, the resulting graph is $(n, 18, 1/3, 2)$ OR-concentrator. Furthermore,* G *has maximum degree d.*

Proof: Let $\mathcal{E}_s$ be the event that a subset of s vertices of L has fewer than cs neighbors in R. For a choice of such a set $S \subseteq L$, and a set T of size cs in R, we have that number of ways to chose a matching such that all the vertices of S has neighbors in T is $cs \cdot (cs-1) \cdots (cs-s+1)$ – indeed, we fix an ordering of the items in S, and assign them their match in T one by one. As such, we have

$$\Xi = \mathbb{P}[\mathcal{E}_s] \leq \binom{n}{s}\binom{n}{cs}\left(\frac{cs(cs-1)\cdots(cs-s+1)}{n(n-1)\cdots(n-s+1)}\right)^d.$$

Using $\frac{cs}{n} \cdot \frac{cs-1}{n-1} \cdots \cdot \frac{cs-s+1}{n-s+1} \leq \left(\frac{cs}{n}\right)^s$, we have

$$\Xi \leq \left(\frac{ne}{s}\right)^s \left(\frac{ne}{cs}\right)^{cs} \left(\frac{cs}{n}\right)^{ds}.$$

The quantity in the right, in the above inequality, is the same quantity bounded in the proof of Theorem 23.1.2, and the result follows by the same argumentation. ■

23.1.2. An expander

Definition 23.1.4. An (n, d, c)***-expander*** is a graph G = (V, E) over n vertices, n, such that

(i) Every vertex in G has degree at most d.

(ii) Any subset S of vertices of V, with $|S| \leq n/3$ has at least $c|S|$ neighbors.

Theorem 23.1.5. *One can construct a $(n, 36, 2)$-expander*

Proof: Let G be a graph with the set of vertices being $[\![n]\!]$. Construction the graph of Theorem 23.1.3, and let G′ be this graph. For every edge $v_i u_j$ in G′ create an edge ij in G. Clearly, G has the desired properties. ■

23.2. Probability Amplification

Let **Alg** be an algorithm in **RP**, such that given x, **Alg** picks a random number r from the range $\mathbb{Z}_n = \{0, \ldots, n-1\}$, for a suitable choice of a prime n, and computes a binary value **Alg**(x, r) with the following properties:

(A) If $x \in L$, then **Alg**$(x, r) = 1$ for at least half the possible values of r.

(B) If $x \notin L$, then **Alg**$(x, r) = 0$ for all possible choices of r.

Next, we show that using $\lg^2 n$ bits[3] one can achieve $1/n^{\lg n}$ confidence, compared with the naive $1/n$, and the $1/t$ confidence achieved by t (dependent) executions of the algorithm using two-point sampling.

Theorem 23.2.1. *For n large enough, there exists a bipartite graph* G(V, R, E) *with $|V| = n$, $|R| = 2^{\lg^2 n}$ such that:*

[3] Everybody knows that $\lg n = \log_2 n$. Everybody knows that the captain lied.

(i) Every subset of $n/2$ vertices of V has at least $2^{\lg^2 n} - n$ neighbors in R.
(ii) No vertex of R has more than $12\lg^2 n$ neighbors.

Proof: Each vertex of V chooses $d = 2^{\lg^2 n}(4\lg^2 n)/n$ neighbors independently in R. We show that the resulting graph violate the required properties with probability less than half.[4]

The probability for a set of $n/2$ vertices on the left to fail to have enough neighbors, is

$$\tau \le \binom{n}{n/2}\binom{2^{\lg^2 n}}{n}\left(1 - \frac{n}{2^{\lg^2 n}}\right)^{dn/2} \le 2^n\left(\frac{2^{\lg^2 n}e}{n}\right)^n \exp\left(-\frac{dn}{2}\frac{n}{2^{\lg^2 n}}\right)$$
$$\le 2^n \underbrace{\left(\frac{2^{\lg^2 n}e}{n}\right)^n}_{*} \exp\left(-\frac{2^{\lg^2 n}(4\lg^2 n)/n}{2}\frac{n^2}{2^{\lg^2 n}}\right) \le \exp\left(n + n\underbrace{\ln\frac{2^{\lg^2 n}e}{n}}_{*} - 2n\lg^2 n\right),$$

since $\binom{n}{n/2} \le 2^n$ and $\binom{2^{\lg^2 n}}{2^{\lg^2 n}-n} = \binom{2^{\lg^2 n}}{n}$, and $\binom{x}{y} \le (\frac{xe}{y})^y$[5]. Now, we have

$$\rho = n\ln\frac{2^{\lg^2 n}e}{n} = n\left(\ln 2^{\lg^2 n} + \ln e - \ln n\right) \le (\ln 2)n\lg^2 n \le 0.7n\lg^2 n,$$

for $n \ge 3$. As such, we have $\tau \le \exp\left(n + (0.7-2)n\lg^2 n\right) \ll 1/4$.

As for the second property, note that the expected number of neighbors of a vertex $v \in R$ is $4\lg^2 n$. Indeed, the probability of a vertex on R to become adjacent to a random edge is $\rho = 1/|R|$, and this "experiment" is repeated independently dn times. As such, the expected degree of a vertex is $\mu\,\mathbb{E}[Y] = dn/|R| = 4\lg^2 n$. The Chernoff bound (Theorem 23.4.1$_{\text{p4}}$) implies that

$$\alpha = \mathbb{P}\left[Y > 12\lg^2 n\right] = \mathbb{P}\left[Y > (1+2)\mu\right] < \exp\left(-\mu 2^2/4\right) = \exp\left(-4\lg^2 n\right).$$

Since there are $2^{\lg^2 n}$ vertices in R, we have that the probability that any vertex in R has a degree that exceeds $12\lg^2 n$, is, by the union bound, at most $|R|\,\alpha \le 2^{\lg^2 n}\exp\left(-4\lg^2 n\right) \le \exp\left(-3\lg^2 n\right) \ll 1/4$, concluding our tedious calculations[6].

Thus, with constant positive probability, the random graph has the required property, as the union of the two bad events has probability $\ll 1/2$. ■

We assume that given a vertex (of the above graph) we can compute its neighbors, without computing the whole graph.

So, we are given an input x. Use $\lg^2 n$ bits to pick a vertex $v \in R$. We next identify the neighbors of v in V: $r_1, \ldots, r_k$. We then compute **Alg**(x, r_i), for $i = 1, \ldots k$. Note that $k = O\left(\lg^2 n\right)$. If all k calls return 0, then we return that **Alg** is not in the language. Otherwise, we return that x belongs to V.

If x is in the language, then consider the subset $U \subseteq V$, such that running **Alg** on any of the strings of U returns TRUE. We know that $|U| \ge n/2$. The set U is connected to all the vertices of R except for at most $|R| - \left(2^{\lg^2 n} - n\right) = n$ of them. As such, the probability of a failure in this case, is

$$\mathbb{P}\left[x \in L \text{ but } r_1, r_2, \ldots, r_k \notin U\right] = \mathbb{P}\left[v \text{ not connected to } U\right] \le \frac{n}{|R|} \le \frac{n}{2^{\lg^2 n}}.$$

We summarize the result.

[4] Here, we keep parallel edges if they happen – which is unlikely. The reader can ignore this minor technicality, on her way to ignore this whole write-up.

[5] The reader might want to verify that one can use significantly weaker upper bounds and the result still follows – we are using the tighter bounds here for educational reasons, and because we can.

[6] Once again, our verbosity in applying the Chernoff inequality is for educational reasons – usually such calculations would be swept under the rag. No wonder than that everybody is afraid to look under the rag.

Lemma 23.2.2. *Given an algorithm* **Alg** *in* **RP** *that uses* $\lg n$ *random bits, and an access explicit access to the graph of Theorem 23.2.1, one can decide if an input word is in the language of* **Alg** *using* $\lg^2 n$ *bits, and the probability of f failure is at most* $n/2^{\lg^2 n}$.

Let us compare the various results we now have about running an algorithm in **RP** using $\lg^2 n$ bits. We have three options:

(A) Randomly run the algorithm $\lg n$ times independently. The probability of failure is at most $1/2^{\lg n} = 1/n$.

(B) Lemma 23.2.2, which as probability of failure at most $1/2^{\lg n} = 1/n$.

(C) The third option is to use pairwise independent sampling (see Lemma 23.4.2$_{\text{p4}}$). While it is not directly comparable to the above two options, it is clearly inferior, and is thus less useful.

Unfortunately, there is no explicit construction of the expanders used here. However, there are alternative techniques that achieve a similar result.

23.3. Oblivious routing revisited

Theorem 23.3.1. *Consider any randomized oblivious algorithm for permutation routing on the hypercube with* $N = 2^n$ *nodes. If this algorithm uses k random bits, then its expected running time is* $\Omega\left(2^{-k}\sqrt{N/n}\right)$.

Corollary 23.3.2. *Any randomized oblivious algorithm for permutation routing on the hypercube with* $N = 2^n$ *nodes must use* $\Omega(n)$ *random bits in order to achieve expected running time* $O(n)$.

Theorem 23.3.3. *For every n, there exists a randomized oblivious scheme for permutation routing on a hypercube with* $n = 2^n$ *nodes that uses* $3n$ *random bits and runs in expected time at most* $15n$.

23.4. From previous lectures

Theorem 23.4.1. *Let* $X_1, \ldots, X_n$ *be n independent variables, where* $\mathbb{P}[X_i = 1] = p_i$ *and* $\mathbb{P}[X_i = 0] = q_i = 1 - p_i$, *for all i. Let* $X = \sum_{i=1}^{b} X_i$. $\mu = \mathbb{E}[X] = \sum_i p_i$. *For any* $\delta > 0$, *we have*

$$\mathbb{P}[X > (1+\delta)\mu] < \left(e^{\delta}\big/(1+\delta)^{1+\delta}\right)^{\mu}.$$

Lemma 23.4.2. *Given an algorithm* **Alg** *in* **RP** *that uses* $\lg n$ *random bits, one can run it t times, such that the runs results in a new algorithm that fails with probability at most* $1/t$, *and uses only* $2\lg n$ *random bits.*

23.5. Bibliographical notes

As usual, we are following here [MR95].

References

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge, UK: Cambridge University Press, 1995.