In this lecture we will see two expander graph based transformations of codes. These constructions can be applied towards obtaining small-biased distributions of improved support size.

Recall that a small-biased distribution over $\{0,1\}^k$ consists of the columns of a generator matrix of a linear code with message length k where the relative hamming weight of every codeword is between $(1-\varepsilon)/2$ and $(1+\varepsilon)/2$.

From the Gilbert-Varshamov bound we know that there exist ε-biased distributions over $\{0,1\}^k$ of (support) size $s = k/H((1-\varepsilon)/2) = O(k/\varepsilon^2)$. In homework 1 you showed that such codes can be found in time $\text{poly}(s)2^s$; this takes a lot of time when k is large and ε is small. Is it possible to do better?

If we are willing to tolerate larger size we can improve this construction by concatenation. Let us recall how this is done. We concatenate an outer Reed-Solomon code (over $\mathbb{F}_{2^m}$) with an inner code meeting the Gilbert-Varshamov bound. To achieve relative distance $1-\varepsilon$, the Reed-Solomon code has rate ε. For the inner code to have relative distance $(1-\varepsilon)/2$, its rate must be $\Theta(\varepsilon^2)$. The concatenated code has relative distance $(1-\varepsilon)^2/2$ and rate $\Theta(\varepsilon^3)$, which corresponds to a $((2-\varepsilon)\varepsilon)$-biased distribution of size $O(k/\varepsilon^3)$.

How long does it take to find such a code? To make this construction work we need to choose $m \approx \log k$, and so the inner code has block length $O(\log n/\varepsilon^2)$. To find such a code we need to invest time $k^{O(1/\varepsilon^2)}$. Can we do even better?

We can improve the efficiency of the construction even further by an additional round of concatenation, but this has the effect of reducing rate from $O(\varepsilon^3)$ to $O(\varepsilon^4)$, and so the resulting ε-biased distribution would have size $O(k/\varepsilon^4)$ and the time to find it would become around $(\log k)^{O(1/\varepsilon^2)}$. Using concatenation alone, it looks like every time we want to improve the efficiency of the construction we have to invest a $O(1/\varepsilon)$ factor in the size of the the small-biased distribution.

1 The ABNNR transformation

Fix a constant $\delta_0 > 0$ and suppose that we have already constructed a linear code C of block length $O(k)$ and relative distance δ_0. When δ_0 is fixed this can be done quite efficiently (say in time $k \cdot \text{poly}\log k$) by the above concatenation-based approach or in other ways.

Alon, Bruck, Naor, Naor, and Roth (ABNNR) give a general transformation that, given any $\varepsilon > 0$, takes a code of block length n and relative distance δ_0 over alphabet $\{0,1\}$ and produces a code of block length n and relative distance $1-\varepsilon$ over alphabet $\{0,1\}^{\text{poly}(1/\varepsilon)}$. Moreover, if the original code is linear, so is the derived one. Applying this transformation to the code C and concatenating with an inner code meeting the Gilbert-Varshamov bound for relative distance $(1-\varepsilon)/2$, we obtain a binary code of rate $O(\varepsilon^3)$ and relative distance $(1-\varepsilon)^2/2 \geq (1-2\varepsilon)/2$. It is easy to check that this code also has maximum distance $(1+\varepsilon)^2/2$, so it gives a 2ε-biased distribution of size $O(k/\varepsilon^3)$.

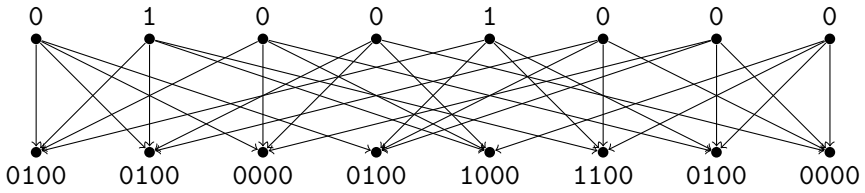
What is the advantage of this construction over the one based on concatenating Reed-Solomon codes? In the previous construction the alphabet size of the inner code depended on k, but now

it only depends on ε. Finding an inner code with the desired properties now takes time $2^{\text{poly}(1/\varepsilon)}$, which is exponential in $1/\varepsilon$ but completely independent of n. The exponent can be made as small as $O(1/\varepsilon^3)$.

To describe this construction we need an n-vertex, d-regular expander graph G. Let $\overline{G}$ be the double cover of G: The graph $\overline{G}$ is a degree-d bipartite graph on $2n$ vertices. Each vertex v of G has two copies v_1, v_2 in $\overline{G}$ and every edge (u, v) of G gives rise to edges $(u_1, v_2), (v_1, u_2)$ in $\overline{G}$.

We now describe the ABNNR transformation that takes a code C of block length n over alphabet $\{0, 1\}$ and produces a code C' of block length n over alphabet $\{0, 1\}^d$. Every codeword $c \in \{0, 1\}^n$ of C gives a codeword $c' \in \{0, 1\}^n$ of C' as follows. We associate a bottom vertex of $\overline{G}$ to every coordinate of c and a top vertex to every coordinate of c'. The value $c'(u)$ of codeword c' at position u is obtained by concatenating the values $c(v_1)c(v_2)\dots c(v_d)$ where $v_1, \dots, v_d$ are the neighbors of u in $\overline{G}$.

Here is an example with $n = 8$, $d = 4$, and $c = \texttt{01001000}$:



Theorem 1. *If C has minimum distance δ_0 then C' has minimum distance at least $1-\varepsilon$, provided $\lambda(G) < \sqrt{\delta_0 \varepsilon}$.*

For simplicity we prove this theorem for linear codes (which is the case we need anyway). Let c be a codeword of C and c' the corresponding codeword in c'. Let S be the set of nonzero coordinates of c and T be the set of zero coordinates of c'. We want to show that it is not possible to have both $|S| \geq \varepsilon n$ and $|T| \geq \delta_0 n$. This follows from a more general statement about expander graphs:

Lemma 2 (Expander mixing lemma)**.** *Let G be a regular graph. For every pair of subsets S and T of vertices of G, $|S| = \alpha n$, $|T| = \beta n$,*

$$\left|\Pr_{\text{edge } (u,v)}[u \in S \text{ and } v \in T] - \alpha\beta\right| \leq \lambda(G) \cdot \sqrt{\alpha\beta}.$$

Since there can be no edges between S and T, the probability in the lemma equals zero, $\lambda(G) \geq \sqrt{\delta_0 \varepsilon}$.

In the last lecture we showed a construction of a 9-regular expander E on n vertices with $\lambda(E) \leq 1-\varepsilon$ for every n that is a square. Let K be a constant that satisfies $\lambda(E)^K \leq 1/9$. Now let $G = E^t$, that is the graph whose edges correspond to paths of length t in E. Then G has degree 9^t and $\lambda(G) = \lambda(E)^t$. If we choose t so that $\lambda(E)^t$ is just below $\sqrt{\delta_0 \varepsilon}$, then Theorem 1 tells us that C' has minimum distance ε. The alphabet size of C' is $2^{9^t} = 2^{1/\lambda(E)^{Kt}} = 2^{O(1/\varepsilon^{2K})}$.

Proof of Lemma 2. Let

$$\mathbf{s}(v) = \begin{cases} 1/\sqrt{n}, & \text{if } v \in S \\ 0, & \text{if } v \notin S \end{cases} \qquad \text{and} \qquad \mathbf{t}(v) = \begin{cases} 1/\sqrt{n}, & \text{if } v \in T \\ 0, & \text{if } v \notin T. \end{cases}$$

We write $\mathbf{s} = \alpha_1\mathbf{v}_1 + \cdots + \alpha_n\mathbf{v}_n$ and $\mathbf{t} = \beta_1\mathbf{v}_1 + \cdots + \beta_n\mathbf{v}_n$ in the basis of eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$ of the normalized adjacency matrix A of G. Then

$$\langle \mathbf{s}A, \mathbf{t}\rangle = \sum_{i,j} \alpha_i\alpha_j\langle \mathbf{v}_iA, \mathbf{v}_j\rangle = \alpha_1\beta_1 + \lambda_2\alpha_2\beta_2 + \cdots + \lambda_n\alpha_n\beta_n$$

from where

$$\big|\langle \mathbf{s}A, \mathbf{t}\rangle - \alpha_1\beta_1\big| \le \lambda(G) \cdot \Big|\sum\nolimits_{i=2}^{n} \alpha_i\beta_i\Big|.$$

We calculate the terms in this expression. First, $\langle \mathbf{s}A, \mathbf{t}\rangle = \sum_{u,v} \mathbf{s}(u)\mathbf{t}(v)A_{u,v}$. There is a contribution to the sum of $1/dn$ for every directed edge (u, v) of G such that $u \in S$ and $v \in T$, so $\langle \mathbf{s}A, \mathbf{t}\rangle$ is exactly the probability that $u \in S$ and $v \in T$ for a random edge (u, v). Second,

$$\alpha_1 = \langle \mathbf{s}, \mathbf{v}_1\rangle = |S|/n = \alpha \quad \text{and} \quad \beta_1 = \langle \mathbf{t}, \mathbf{v}_1\rangle = |T|/n = \beta.$$

Finally,

$$\Big\|\sum\nolimits_{i=2}^{n} \alpha_i\beta_i\Big\| \le \sqrt{\sum_{i=2}^{n} \alpha_i^2} \cdot \sqrt{\sum_{i=2}^{n} \beta_i^2}.$$

Since the basis $\mathbf{v}_1, \ldots, \mathbf{v}_n$ is orthonormal, $\sum_{i=2}^{n} \alpha_i^2 = \|\mathbf{s}\|^2 - \alpha_1^2 = \alpha - \alpha^2$ and similarly $\sum_{i=2}^{n} \beta_i^2 = \beta$. This gives the slightly stronger inequality

$$\big|\Pr_{\text{edge } (u,v)}[u \in S \text{ and } v \in T] - \alpha\beta\big| \le \lambda(G) \cdot \sqrt{\alpha(1-\alpha) \cdot \beta(1-\beta)}. \qquad \square$$

2 A different way to improve the bias via expanders

We now show an alternative construction of ε-biased distributions of size $O(k/\varepsilon^K)$ for some fixed constant K. Unlike in the ABNNR transformation K here will be larger than 3 (it seems the best it can do is $K = 4$), but the distributions can be constructed in time polynomial in k *and* $1/\varepsilon$. This is an unpublished construction of Eyal Rozenman and Avi Wigderson (and maybe others).

As in the ABNNR approach, this construction starts with an ε_0-biased distribution D_0 over $\{0,1\}^k$ of size $O(k)$ and some fixed bias $\varepsilon_0 < 1/2$. (Any $\varepsilon_0 < 1$ will suffice but let us make the stronger assumption for simplicity.) Again we use expander graphs to improve the bias of the distribution at the expense of making it larger, but the expanders will now be used in a different way. What we need is the following statement which you will prove in homework 3:

Theorem 3. *Let $D \subseteq \{0,1\}^n$ be an ε-biased distribution and G be a regular graph whose vertices are labeled by samples of D so that the number of vertices labeled x is proportional to the probability of x under D. Let D' be the following distribution: Uniformly choose a random edge (x_1, x_2) of G and output $x_1 + x_2$. Then D' is $(\varepsilon^2 + \lambda(G))$-biased.*

We apply this construction iteratively. Suppose D_i is an ε_i-biased distribution of size s_i. We apply the theorem on D_i and a graph G_i of degree d_i and $\lambda_i = \lambda(G_i)$. Then the resulting distribution D_{i+1} has bias $\varepsilon_{i+1} = \varepsilon_i^2 + \lambda_i$ and size $s_{i+1} = d_is_i$ (the elements of D_{i+1} correspond to ordered edges of G_i).

We now choose G_i so that $\lambda_i = \varepsilon_i^2$ and $d_i = 1/\lambda_i^K$. (As above G_i can be chosen as a fixed power of the expander from last lecture.) This gives the recursive relation

$$\varepsilon_{i+1} = 2\varepsilon_i^2 \quad \text{and} \quad s_{i+1} = s_i/\varepsilon_i^{2K}$$

with $\varepsilon_0 < 1$, $s_i = O(n)$. Solving this recurrence gives $\varepsilon_i = (2\varepsilon_0)^{2^i}/2$ and

$$s_i = \frac{O(n)}{(\varepsilon_0\varepsilon_1 \dots \varepsilon_{i-1})^{2K}} = \frac{O(2^{2Ki}n)}{(2\varepsilon_0)^{2K(2^i-1)}} \leq \frac{O(n)}{(2\varepsilon_0)^{2K\cdot(2^i+i)}} = O(n/\varepsilon_i^{2(1+i/2^i)K}) = O(n/\varepsilon_i^{4K}).$$