Lecture 5: Expander Codes

Anup Rao

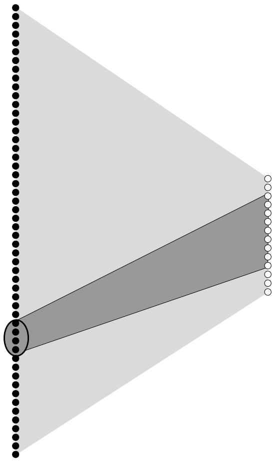
October 9, 2019

The main drawback of Reed-Solomon codes is the large alphabet size. Expander codes are codes that do not have this drawback. The properties of expander codes follow from the combinatorial properties of graphs called *expander graphs*.

An expander graph is a sparse graph with the property that every (small enough set) S expands — namely the neighborhood of S in the graph is larger than S itself. For the purpose of building an error-correcting code, it is best to start with a bipartite expander graph. Recall that a graph is bipartite if the vertices can be partitioned into sets U, V such that all edges go between U and V.

Expander graph have been the subject of extensive study in mathematics and computer science in the last 50 years. They have many many applications, this is just one.

Definition 1. *A bipartite graph with bipartition U, V is called an (α, β) expander, if for every set $S \subseteq U$ of size at most $\alpha \cdot |U|$, the number of vertices in V that are connected to S is at least $\beta \cdot |S|$.*



Let G be a bipartite graph as above, with $|U| = n, |V| = m$, such that every vertex in U has exactly $D > 2$ neighbors, and the graph is a (α, β) expander, with $\beta > 3D/4$. These parameters ensure a strong property: not only does every set contained in U expand, but it will have many *unique neighbors*. Given $S \subseteq U$, say that $v \in V$ is a unique neighbor of S if v exactly 1 neighbor in S.

Lemma 2. *If G is as above, then every $S \subseteq U$ of size $|S| \leq \alpha n$ has more than $(D/2) \cdot |S|$ unique neighbors.*

Proof. Suppose S has u unique neighbors. Then by counting the number of edges emanating from S, we get

$$D \cdot |S| \geq u + (\beta|S| - u) \cdot 2,$$

which gives $u \geq 2\beta - D > D/2$, upon rearranging. □

We shall use this graph to construct an error correcting code. The code is extremely simple to describe: let H be the $m \times n$ matrix defined by

Later, we shall prove that such expander graphs exist.

$$H_{i,j} = \begin{cases} 1 & \text{if } (j,i) \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}.$$

Then the code code C is given by

$$C = \{x \in \mathbb{F}_2^n : Hx = 0\}.$$

These codes are also called *low-density parity check codes*, because the parity check code H is a sparse matrix. Since we have added m linear constraints, the dimension of the code is at least $n - m$.

Let us start by analyzing the distance of the code:

Lemma 3. *The distance of C is at least αn.*

Proof. Since the code is linear, it suffices to show that every codeword has at least αn ones. Indeed, suppose this is not the case, and $x \in C$, but x has less than αn ones. Let $S \subseteq U$ be the set of vertices in the graph that correspond to the 1's of x. Since S has more than $D/2$ unique neighbors, we see that at least $D/2$ of the parity checks are non-zero. □

So, we have a bona fide code. The truly amazing thing is that the decoding algorithm for the code is extremely simple as well. To decod a given a received word y to a codeword repeat the following steps until there is nothing left to do — if there is a vertex on the left such that the majority of its neighbors have non-zero parity check, flip the value of the vertex. That is the whole algorithm!

In fact, you can prove that you can do the flipping *in parallel*, giving an even faster decoding algorithm.

Lemma 4. *The above decoding algorithm recovers any codeword from up to $\alpha n/2$ errors.*

Proof. First observe that throughout the algorithm, the number of erroneous parity checks never increases. This already implies that when the algorithm stops, it must terminate with the correct codeword. Indeed, we start at distance at most $\alpha n/2$ errors, which means that the number of unsatisfied parity checks is at most $\alpha Dn/2$. If at some point, we are working with a word at distance αn from the correct codeword, then Lemma 2 implies that the number of incorrect parity checks is greater than $(D/2)\alpha n$. This is impossible, since the number of incorrect parity checks cannot increase.

The only remaining thing to check is that there will always be a vertex in U that the algorithm can pick in each step. This again follows from Lemma 2. At each point, if the coordinates with errors is denoted S, then S has at most $D|S|$ neighbors, but more than half of them are unique neighbors. So, so some vertex of S must see more than $D/2$ unique neighbors. This vertex is a candidate for the algorithm to flip.

□

It only remains to prove that expanders with the above properties can be constructed. This is a major project, involving many beautiful ideas that we do not discuss here. We restrict our attention to showing that such expanders exist.

For a parameters D, α, let $m = 8e^2 D\alpha n$. Pick a uniformly random graph by connecting each of the n vertices in U to D vertices in V. Let us compute the probability that the graph is not an expander. Whenever the graph does not expand, there must be a set $S \subseteq U$ of size $k \le \alpha n$ such that the kD edges coming out of S had lots of collisions. Specifically, if we imagine choosing these kD edges one by one, $D/4$ of them must have landed among the previously chosen ones.

The probability of this event is at most

$$\begin{aligned}
&\binom{n}{k} \cdot \binom{kD}{kD/4} \cdot \Big(\frac{kD}{m}\Big)^{kD/4} \\
&\le \Big(\frac{en}{k}\Big)^{k} \cdot \Big(\frac{4ekD}{kD}\Big)^{kD} \cdot \Big(\frac{kD}{m}\Big)^{kD/4} && \text{using } \binom{n}{k} \le \frac{en}{k} \\
&= \Big(\frac{en/k}{(m/(4ekD))^{D/4}}\Big)^{k} \\
&= \Big(\frac{en/k}{(2\alpha en/k)^{D/4}}\Big)^{k}
\end{aligned}$$

If we choose $D = O(\log(1/\alpha))$, the above quantity is at most $1/4^k$. This means that the probability that the graph is not an expander is at most $\sum_{k=1}^{\infty} 1/4^k = 1/2$.

The result is a code that recover from an error rate of $\alpha/2$, and has rate $1 - O(\alpha \log(1/\alpha))$. For small α, this expression looks like $1 - O(h_2(\alpha))$. Recall that the Hamming bound says we cannot beat rate $1 - h_2(\alpha)$.

Expanders from Eigenvalues

THE EXPANDER CODES we saw in the last section required very strong expanders, whose expansion is almost D. These are quite hard to construct. However, it turns out that one can improve the construction so that even weak expansion suffices.

In order to carry out the analysis in this section, we need a *spectral* definition of expanders. The crash course in spectral expanders starts now.

Every real valued matrix A can be written as

$$A = U^{\intercal} S V,$$

where U, V are unitary matrices, and S is a diagonal matrix. This is called the *singular value decomposition* of A, and the entries on the diagonal of S are the singular values of A.

When A is symmetric, for example if it is the adjacency matrix of

Recall that an $N \times N$ unitary matrix is a matrix that encodes a rotation. In other words, its columns and rows both form orthonormal bases. In other words $VV^{\intercal} = V^{\intercal}V = I$.
So, to compute Ax, first rotate x with Vx, then scale the coordinates using S, then rotate again using $U^{\intercal}$.

an undirected graph, then we have

$$A = V^{\mathsf{T}} E V,$$

where V is unitary, and E is diagonal. The entries of E are now the eigenvalues of A, and the rows of V are the eigenvectors.

Given a D-regular graph on N vertices, with adjacency matrix A, it is easy to see that the eigenvalues of A all lie in between D and $-D$, and there is an eigenvalue of D that corresponds to the eigenvector $(1/\sqrt{N}, \ldots, 1/\sqrt{N})$.

We say that the graph is an expander if the second largest magnitude eigenvalue (namely the second entry in the list of eigenvalues sorted by absolute value) is bounded away from D. A key fact is the following lemma:

Lemma 5 (Expander mixing). *Suppose G is a D-regular graph with $N \times N$ adjacency matrix A with second largest magnitude eigenvalue λ. Suppose S, T are subset of vertices. If e denotes the number of edges between S and T, where we count edges in $S \cap T$ twice, then*

$$\left| e - \frac{D \cdot |S| \cdot |T|}{N} \right| \leq \lambda \sqrt{|S| \cdot |T|}.$$

Note that $D|S||T|/N$ is the number of edges that you would expect to lie between S, T in a random D-regular graph. So, the lemma asserts that the graph behaves like a random graph.

Can you use the expander mixing lemma to prove that every set of linear density must *expand* in the sense of the previous section?

Similarly, we have:

Lemma 6 (Bipartite Expander mixing). *Suppose G is a D-regular bipartite graph with N vertices on the left and right, $N \times N$ symmetric adjacency matrix A, and with second largest magnitude eigenvalue λ. Suppose S, T are subset of vertices. If e denotes the number of edges between S and T, then*

$$\left| e - \frac{D \cdot |S| \cdot |T|}{N} \right| \leq \lambda \sqrt{|S| \cdot |T|}.$$

Proof. Let $1_S, 1_T$ be the indicator vectors for the sets S, T. Suppose

$$A = \begin{bmatrix} 1^N/\sqrt{N} \\ W \end{bmatrix}^{\mathsf{T}} \cdot \begin{bmatrix} D & & \\ & \pm\lambda & \\ & & \ddots \end{bmatrix} \cdot \begin{bmatrix} 1^N/\sqrt{N} \\ W \end{bmatrix}$$

is the eigenvalue decomposition. Then we have $e = 1_S^{\mathsf{T}} \cdot A \cdot 1_T$, so

$$e - D \cdot \frac{|S|}{\sqrt{N}} \cdot \frac{|T|}{\sqrt{N}} = 1_S^{\mathsf{T}} \cdot W^{\mathsf{T}} \cdot \begin{bmatrix} \pm\lambda & \\ & \ddots \end{bmatrix} \cdot W \cdot 1_T.$$

In other words $1_S^{\mathsf{T}} A 1_T$ corresponds to projecting $1_S, 1_T$ onto the basis of eigenvectors, multiplying these vectors coordinate-wise in this basis, and taking a weighted average of the resulting vector using the eigvenvalues. In the above expression, we took out the contribution of the top eigenvalue. All other terms are scaled by an eigenvalue of magnitude at most λ.

Since all entries on the diagonal are at most λ in magnitude we get:

$$\begin{aligned} \left| e - D \cdot \frac{|S| \cdot |T|}{N} \right| &\leq \lambda \cdot 1_S^\intercal \cdot W^\intercal \cdot W \cdot 1_T \\ &\leq \lambda \sqrt{|S| \cdot |T|}, \end{aligned}$$

where the second inquality follows from Cauchy-Schwartz. □

To see why the expander mixing lemma implies expansion, suppose S a set of at most αN vertices on the left, and let T be the set of neighbors of S. By construction, $|T| \leq \alpha D N$. Then the number of edges between S, T is $D \cdot |S|$. So, the expander mixing lemma asserts that

$$\begin{aligned} &D \cdot |S| - D \cdot \frac{|S| \cdot |T|}{N} \leq \lambda \sqrt{|S| \cdot |T|} \\ \Rightarrow &D(1 - \frac{|T|}{N}) \leq \lambda \sqrt{\frac{|T|}{|S|}} \\ \Rightarrow &D(1 - \alpha D) \leq \lambda \sqrt{\frac{|T|}{|S|}}. \end{aligned}$$

Set α so that $1 - \alpha D \geq 1/2$. Then we get $|T|/|S| \geq (D/(2\lambda))^2$.