CS6845: Pseudorandomness 5 Mar 2018

24. Expander codes - Linear-time decoding

24.1 Introduction

In the last lecture, we saw the construction codes from expander graphs, and proved that good expansion implies large distance. In this lecture, we will give a linear time decoding algorithm for expander codes. We will also define the notion of lossless expanders, which are used to construct asymptotically good expander codes.

24.2 Expander codes

We know that random bipartite graphs are good expanders for certain parameters. In this section we will see how to use the properties of the expander to construct good codes. We will not prove that explicit expanders with the properties that we require can be constructed. Our aim will be to analyse the codes coming out of these expanders, assuming good explicit constructions.

Let $H \in \mathbb{F}_2^{n\times(n-k)}$ be the parity check matrix of a code $\mathcal{C}$. We can think of H also as a bipartite graph (V_L, V_R, E) where $|V_L| = n$ and $|V_R| = n-k$. The number of parity check constraints in which each $i \in V_L$ is involved is the *left degree* of the graph. We will look at bipartite graphs that are D-left regular.

For this discussion we want expanders, where for every $S \subseteq V_L$ such that $|S| \leq d$, $|N(S)| \geq L(H,d)|S|$, where $L(H,d)$ is known as the left-expansion ratio. For the remainder of this discussion the set $N(S)$ will denote the neighbors of S excluding the set S. In the last lecture we proved the following theorem.

Theorem 24.1 (Sipser & Spielman)**.** *If $L(H,d) > D/2$, then the minimum distance of $\mathcal{C}$ is $> d$.*

Now we move into the description and analysis of the decoding algorithm.

24.2.1 Decoding algorithm

We have shown that if the parity check matrix is the bipartite adjacency matrix of an expander, then the code has good distance. Now we will describe the decoding algorithm. The idea of the algorithm is the following: Suppose $\mathbf{x}$ is the received word, and suppose that $\mathrm{wt}(\mathbf{x}+e_i)H < \mathrm{wt}(\mathbf{x}H)$, then we can flip the bit i. We will show that if the code has distance $> d$, then we can correct $d/2$ errors after $O(n-k)$ iterations of the above step.

What does it mean to say that $\mathrm{wt}(\mathbf{x}+e_i)H < \mathrm{wt}(\mathbf{x}H)$? For the vertex $i \in V_L$, the set $N(i)$ are the parity check equations to which it is part of. Saying that $\mathrm{wt}(\mathbf{x}+e_i)H < \mathrm{wt}(\mathbf{x}H)$ means that the number of parity checks that are unsatisfied in $N(i)$ is strictly greater than the number of parity checks that are satisfied in $N(i)$. Only then can the flipping of the i^{th} make the Hamming weight smaller. We want to show that no matter how we choose the bit to flip, this process cannot go on forever. We now state the correctness proof the decoding algorithm.

Theorem 24.2 (Sipser & Spielman). *Let H be a D-regular bipartite graph such that $L(H,d) > \frac{3}{4}D$. Let $\mathbf{y} \in \{0,1\}^n$ be such that $\Delta(\mathbf{y},\mathcal{C}) \leq d/2$. Then the decoding algorithm above ends after $O(n-k)$ many iterations.*

Proof. Let $\mathbf{x} \in \mathcal{C}$ and let $\Delta(\mathbf{x},\mathbf{y}) \leq d/2$. Let $\mathbf{y}^i$ denote the vector $\mathbf{y}$ after the i^{th} iteration of the algorithm, and let $A_i = \{u \mid \mathbf{y}^i_u \neq \mathbf{x}_u\}$. Assume that $A_i \neq \emptyset$ and $|A_i| \leq d$.

Let $U_i \subseteq N(A_i)$ be the set of constraints not satisfied, and let S_i be the set of constraints that are satisfied. Then, $|U_i| + |S_i| = |N(A_i)| > \frac{3}{4}D|A_i|$. Also each constraint in S_i will have at least two neighbors in A_i (why?). Therefore, $|U_i| + 2|S_i| \leq D|A_i|$. So we have $|U_i| > D|A_i|/2$. Therefore, there exists some $v \in A_i$ such that $|N(v) \cap U_i| > D/2$ and we can find a variable to flip. Notice that each step we are reducing the number of unsatisfied constraints, so this algorithm will terminate in $O(n-k)$ many steps.

Now we show that $|A_i| < d$ for every i. This makes sure that we don't move into a wrong codeword during the decoding process. First observe that A_i increases or decreases by at most 1 from the previous step. Therefore, suppose that at some point $|A_i| = d$. Then, we know that $|U_i| > Dd/2$. Since $|A_0| \leq d/2$, we have that $|U_0| \leq |N(A_0)| = Dd/2$. But this is not possible since U_i cannot increase at any stage of the algorithm. □

To obtain a linear time decoding algorithm we first mark which of the constraints are not satisfied. This takes $O((n-k)t)$ time, where t is maximum degree among the nodes in V_R. Now for each vertex $v \in V_L$, we count the number of unsatisfied parity checks it is part of. This takes $O(nD)$ time. While doing this we add the vertices that have $> D/2$ unsatisfied constraints into a queue Q. Now we dequeue Q, and for each v, we flip the corresponding bit. When we do that we mark the constraints that change from satisfied to unsatisfied and unsatisfied to satisfied. This takes $O(D)$ time. Since at most D constraints are affected, and the right-degree is t, we can modify the queue and add new elements to it in time $O(Dt)$. Now the entire process runs for at most $n-k$ iterations. Therefore, the total running time is $O(Dt(n-k)) = O(n)$ since D is a constant.

Lossless expanders

To obtain the explicit asymptotically good code, we need an explicit construction of the expanders. Capalbo, Reingold, Vadhan and Wigderson provide explicit constructions of *lossless expanders* that achieve the bounds required to construct the codes.

Definition 24.3 (Lossless expanders). *A left D-regular bipartite graph $G(V_L, V_R, E)$ where $|V_L| = m$ and $|V_R| = n$ is a (K,ε)-lossless expander if for every $S \subseteq V_L$ such that $|S| \leq K$, $|N(S)| \geq (1-\varepsilon)D|S|$.*

Theorem 24.4 (Capalbo, Reingold, Vadhan, Wigderson). *For any $\varepsilon > 0$ and $m \leq n$, there is an explicit family of left D-regular bipartite graphs that are $(\Omega(\varepsilon m/D), \varepsilon)$-lossless expanders, where $D \leq (n/\varepsilon m)^c$.*

If we use these expanders in our code construction with $m = \alpha n$ for some $\alpha < 1$, we get an $(\varepsilon \alpha n/D, \varepsilon)$-lossless expander with $D \leq (\alpha\varepsilon)^{-c}$. In our construction $k = n - m = \Omega(n)$ and $L(H, d) > .99D$ for $d = \Omega(n)$. Thus, we have an asymptotically good family of codes with linear time decoding.