Coding Theory
Lecture 8

Lecture by Dr. Elette Boyle
Typeset by Steven Karas

2017-12-26
Last edited 15:43:48 2017-12-28

Disclaimer These lecture notes are based on the lecture for the course Coding Theory, taught by Dr. Elette Boyle at IDC Herzliyah in the fall semester of 2017/2018. Sections may be based on the lecture notes written by Dr. Elette Boyle.

Agenda

- An efficient class of binary codes (expander codes)
  - Today - construction + distance analysis
  - Tomorrow - efficient decoding (linear time)

1 Review & Background

Last week, we covered an efficient decoding algorithm for RS codes.

RS codes in general have good distance and efficient decoding, but they rely on a large alphabet $\mathbb{F}_q$.

Concatenated codes is a way to compose codes on top of each other, which is a typical strategy in implementing RS codes.

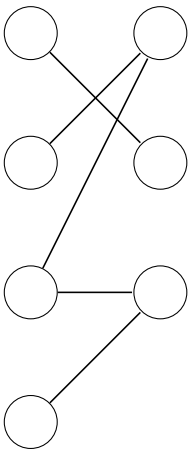
1.1 Bipartite graphs

A bipartite graph is a triple $G = (L, R, E)$ of left vertices L, right vertices R, and edges $E \subseteq L \times R$.

1.2 Adjacency matrix

The adjacency matrix of a graph is defined as A_G where the (u, v) entry of A_G is 1 if $(u, v) \in E$.

Adjacency of bipartite graphs Because only a small subset of the generic adjacency matrix of a bipartite graph is populated, we can simplify it to be a $|R| \times |L|$ matrix.



$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

1.3 Factor Graph

The adjacency matrix can be viewed as a parity-check matrix of a linear code. Similarly, any binary linear code can view the parity check matrix as the adjacency matrix of a bipartite graph. Note that the parity check matrix must be full rank, or the code is overly restrained with no codewords.

Specifically, given a parity check matrix H, we can construct a bipartite graph G_H with adjacency matrix H.

Or, given a bipartite graph $G = (L, R, E)$ where $|L| \geq |R|$, we can construct a linear code $C(G)$ with parity check matrix A_G.

2 Expander Codes

2.1 Expander Graphs

Informally, sparse graphs with good connectivity properties.

We will consider bipartite graphs with a linear number of edges (with relation to the number of vertices).

Left regularity A bipartite graph $G = (L, R, E)$ is D-regular if every vertex in L has degree exactly D.

Neighbor set For any vertex set $S \subseteq L$, $u \in R$ is a neighbor of S if $\exists s \in S : (s, u) \in E$. We denote by $N(S)$ the set of all neighbors of S, noting that $N(S) \subseteq R$.

Unique Neighbor Set For any $S \subseteq L$, $u \in R$ is a unique neighbor of S if there is exactly one edge that connects u to S. Denote $U(S)$ as the set of all unique neighbors of S.

Bipartite Expander A bipartite graph where all subsets of L up to some size where $|U(S)| > |S|$.

Formally, an $(n, m, D, \gamma, \alpha)$ bipartite expander is a D-regular bipartite graph $G = (L, R, E)$ where $|L| = n$, $|R| = m$ such that for any $S \subseteq L$, it holds that $|S| \leq \gamma n$ and $|N(S)| \geq \alpha |S|$.

As such, $\frac{1}{n} \leq \gamma \leq 1$. Note that we trivially get that $(n, m, D, \frac{1}{n}, D)$ is an expander because each subset trivially expands to D vertices. Similarly, $0 \leq \alpha \leq D$.

Note that γ gives a measure of how small the expanding sets are, where a bigger γ is stronger. Similarly, α is the expansion factor which gives a measure of how much the sets expand.

Existence We can achieve expansion arbitrarily close to D.

For any $\varepsilon > 0$ and $n \geq m$, there exists some bipartite expander $(n, m, D, \gamma, D(1-\varepsilon))$ where $D = \Theta\left(\frac{\log(n/m)}{\varepsilon}\right)$ and $\gamma = \Theta\left(\frac{\varepsilon m}{Dn}\right)$.

The expansion factor $\alpha = D(1-\varepsilon)$ can get arbitrarily close to D, at the expense of a larger degree $D = \Theta\left(\frac{\log(n/m)}{\varepsilon}\right)$.

Note that we constrain this to $n \geq m$. It is trivial to construct such an expander for the case where $m = Dn$ by simply mapping to disjoint sets in R.

A trivial lower bound on an achievable m is $\gamma n D(1-\varepsilon)$ as sets of size γn must expand by a factor of α. Our proof will show for an $m \sim \frac{1}{\varepsilon}$ times longer.

Lemma - Useful Property Let $G = (L, R, E)$ be a $(n, m, D, \gamma, D(1-\varepsilon))$ bipartite expander with $\varepsilon < 1/2$.

We want to show that for any $S \subseteq L$ of size $|S| \leq \gamma n$, it holds that $|U(S)| \geq D(1-2\varepsilon)|S|$.

Recall that $|N(S)| \geq D(1-\varepsilon)|S|$. Moreover, D-regularity gives us that the total number of edges between S and $N(S)$ is exactly $D|S|$.

We use $|N(S)|$ edges to touch the neighbors (or they wouldn't be neighbors). This gives us at most $D|S| - D(1-\varepsilon)|S| = \varepsilon D|S|$ remaining edges. In the worst case, each remaining edge disqualifies a different vertex, which gives us $|U(S)| \geq N(S) - \varepsilon D|S| = D(1-2\varepsilon)|S|$.

2.2 Expander Codes

Let $G = (L, R, E)$ be a bipartite expander graph where $|L| \geq |R|$. Then $C(G)$ is an expander code.

Note that for $|L| = n$ and $|R| = n - k$, then for $(c_1, \ldots, c_n) \in \{0,1\}^n$ is a codeword in $C(G)$ iff $\forall r \in R$:

$$\sum_{l \in S : r \in N(\{l\})} c_l = 0$$

We've already shown that the rate of $C(G) = \frac{n-k}{n} = \frac{|L|-|R|}{|L|}$.

We want to show the distance.

Weak distance bound Let G be a $(n, m, D, \gamma, \overbrace{D(1-\varepsilon)}^{\alpha})$ bipartite expander where $\varepsilon < 1/2$. We claim that C_G is a $[n, k, \gamma n + 1]_2$ code.

We get all our claims for free except that the distance is $\gamma n + 1$. Suppose for contradiction that the distance of $C(G)$ is $\leq \gamma n$. Because C_G is a linear code, there exists some codeword $\vec{c} \in C(G)$ with weight $\leq \gamma n$. Let $S = \{l \in L \mid c_l \neq 0\}$ (the vertices where the codeword is nonzero). As such, $|S| = w(\vec{c}) \leq \gamma n$. Because G is an expander, by lemma, it holds that:

$$|U(S)| \geq \underset{>0}{D} \quad \underset{>0;\varepsilon<\frac{1}{2}}{(1-2\varepsilon)} \quad \underset{>0;\vec{c}\neq\vec{0}}{|S|}$$

So there exists some vertex $r \in R$ with $r \in U(S)$. However, this implies that the parity of r is 1, because S is the set of all left vertices which are 1, and there is exactly one edge to r. However, this would imply that the parity check defined by r is not satisfied by $\vec{c}$ (as $c_l \neq 0$). Therefore, $\vec{c} \notin C(G)$. This contradicts our assumption that the distance of $C(G)$ is $\leq \gamma n$, and therefore the distance is $\geq \gamma n + 1$.

A better distance bound Instead of considering codewords of weight only up to the expansion threshold of γn, we will leverage expansion of a subset of the corresponding nonzero positions S.

Let G be a $(n, n-k, D, \gamma, D(1-\varepsilon))$ bipartite expander with $\varepsilon < 1/2$. We claim that $C(G)$ has distance of at least $2\gamma(1-\varepsilon)n$.

Suppose for contradiction that the distance is strictly less than $2\gamma(1-\varepsilon)n$. Therefore, there is a nonzero codeword $\vec{0} \neq \vec{c} \in C(G)$ with weight $w(\vec{c}) < 2\gamma(1-\varepsilon)n$.

Let $S = \{l \in L \mid c_l \neq 0\}$ be the nonzero vertices of $\vec{c}$. We will again argue that $U(S) \neq \emptyset$ and then use the above argument to contradict.

If $|S| \leq \gamma n$ then the previous theorem still holds. So assume that $T \subset S$ such that $|T| = \gamma n$. Therefore, $|U(T)| \geq D(1-2\varepsilon)|T|$. But, some of the vertices in $U(T)$ may also have a neighbor in $S \setminus T$. We want to upper bound the number of neighbors of $S \setminus T$. Since the number of edges from $S \setminus T$ is $D|S \setminus T|$, it follows that:

$$\begin{aligned} |N(S \setminus T)| &\leq D|S \setminus T| \\ &= D(|S| - |T|) \\ &\leq D(2\gamma(1-\varepsilon)n - \gamma n) \\ &= D(\gamma(1-2\varepsilon)n) \end{aligned}$$

Note that this implies that $|U(S)| \geq \underset{\geq D(1-2\varepsilon)\gamma n}{|U(T)|} - \underset{<D(1-2\varepsilon)\gamma n}{|N(S \setminus T)|}$ (as we throw away any $r \in U(T)$ with an additional edge to $S \setminus T$). This implies that $|U(S)| > 0$, from which we can repeat the argument from before.

2.3 Encoding & Decoding

Because this is a linear code, we can simply find the generator matrix which gives us encoding in $O(n^2)$ time by matrix multiplication. For expander codes, we can do this in linear time[1].

Decoding can be done in linear time because of the sparsity of the parity check matrix.

Decoding at a high level Given a graph $G = (L, R, E)$ and a received word $\vec{y} \in \{0,1\}^n$.

First, evaluate the right hand side parity checks. Some will be satisfied and some unsatisfied.

Next, for every $l \in L$, count how many of its D neighboring checks are unsatisfied.

Until there does not exist $l \in L$ with $> \left\lfloor \frac{D}{2} \right\rfloor$ neighboring checks unsatisfied, flip y_l.

Because in each iteration, the corresponding right side neighbors flip from satisfied to unsatisfied and vice versa.

3 Next Time

Proof of decoding convergence. Proof of decoding correctness. Linear time encoding.

References

[1] Michael Sipser and Daniel A Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.