CSE522/Math581EB Coding Theory **Fall 2001**

Lecture 10: November 8

Lecturer: Elchanan Mossel *Scribe: Justin Campbell*

10.1 Linear Encoding and Decoding

We've seen good codes with linear time decoding, but the encoding was not linear. We now turn to achieving this goal. We follow [Spi].

10.1.1 Error reduction via Expanders

Once again we will use expander graphs.

Definition 10.1 *We begin with a bipartite graph $G = (L, R, E)$ where the vertices are $\{1, \ldots, n\}$, L and R form the two "sides" of the graph and $|L|+|R| = n$. We form the code C_G from all $x = (x_1, \ldots, x_n) \in \{0,1\}^n$ such that for $r \in R$ with neighbors $l_1, \ldots, l_j \in L$, we have $x_r = x_{l_1} + \cdots + x_{l_j}$.*

We are looking at a code formed from a graph where the i-th bit of the binary n-vector corresponds to the vertex i. The value of a bit corresponding to a vertex in R is the sum of the values of its neighbors in L. We think of L as the message bits and R as the check bits. The encoding is done in linear time. In addition, the distance is at most $d_L + 1$, since if we flip a bit on the left we flip all of its neighbors. But what about decoding?

Definition 10.2 *A is an (ϵ, γ)-error-reducing algorithm if $\forall (m, c) \in C_G$ and $(x, y) \in \{0,1\}^n$ such that if $\Delta(x, m) = s \leq \gamma n$ and $\Delta(y, c) = t \leq \gamma n$ then A returns x' such that $\Delta(x', m) \leq \epsilon t$.*

(m, c) is notational shorthand for splitting a codeword into two parts. We think of a codeword $(m, c) \in C_G$ as the concatenation of m with c, where m are the message bits and c the check bits. We first give a simple algorithm.

Definition 10.3 *The 'Naive' error-reduction algorithm*

while $\exists l \in L$ with
more unsatisfied neighbors then satisfied neighbors
flip x_l

This is a linear-time algorithm (by a similar argument to the one used for the linear time decodable codes).

Theorem 10.4 *If G is a $(d, 2d)$-regular graph and a $(\frac{7}{8}d, \delta)$-expander then the 'Naive' algorithm is an $(\epsilon = \frac{4}{d}, \gamma = \frac{\delta d}{2(d+2)})$-error-reducing algorithm.*

Proof: We use a few definitions to help the argument:

$S \subseteq L = \{i \mid x_i \neq m_i\}$,
$T \subseteq R = \{j \mid y_j \neq c_j\}$,
$U = \{j \in R \mid j\text{-th bit unsatisfied }\}$,
$A = \{j \in \partial(S) \mid j \text{ has exactly one neighbor in } S\}$.

Remember that ∂ is the boundary of a set, i.e. all of the neighbors of elements in S. Lastly we define:

$B = \{j \in \partial(S) \mid j \text{ has } > 1 \text{ neighbors in } S\}$

We use the following claims to prove the theorem.

Claim 10.5 $A \setminus T \subseteq U \subseteq \partial(S) \cup T$

The left containment comes from the definitions. The right containment comes because either x_j is bad (so in T) or there is an l_i that's bad.

Claim 10.6 *Let* $|U| = u, |S| = s, |T| = t$ *then* $u \leq ds + t$.

This follows from the previous claim and the regularity of the graph.

Claim 10.7 *If* $\delta k > s \geq 0$ *then* $|U| \geq |A \setminus T| \geq (2\alpha - d)s - t$ *where* $\alpha = \frac{7}{8}d$.

$|U| \geq |A \setminus T| \geq |A| - |T| \geq |A| - t$ and from the previous lecture $|A| \geq (2\alpha - d)s = \frac{7}{4}d - ds - t$.

Claim 10.8 *Assuming the suppositions stated in the theorem, where* $\gamma = \frac{\delta d}{2(d+2)}$ *then at all steps of the algorithm it holds that* $s < \delta k$.

Initially: $s < \gamma n < \delta k$. Now since $k \approx \frac{2}{3}n$ and $\frac{d}{2(d+2)} \approx \frac{1}{2}$, and from an earlier claim we have that $u \leq ds + t < (d+1)\gamma n$.

Since u decreases we see that $u \geq (2\alpha - d)s - t \Rightarrow s \leq \frac{u+t}{2\alpha - d} < \frac{(d+1)\gamma n + \gamma n}{2\alpha - d} = \frac{(d+2)\gamma n}{2\alpha - d} < \frac{(d+2)\gamma n}{\frac{3}{4}d} = \frac{(d+2)\frac{\delta d}{2(d+2)}n}{\frac{3}{4}d} = \frac{2}{3}\delta n = \delta k$. Lastly,

Claim 10.9 *If* $s > \frac{4t}{d} = t\epsilon$ *then* $\exists i \in S$ *such that* i *has more than* $\frac{d}{2}$ *neighbors in* U.

$|A \setminus T| \geq (2\alpha - d)s - t = \frac{3}{4}ds - t \geq \frac{3}{4}ds - \frac{ds}{4} > \frac{ds}{2}$. Since $A \setminus T \subseteq U$ and all $j \in A \setminus T$ have some neighbors in S, there $\exists i \in S$ with more than $\frac{d}{2}$ neighbors in $A \setminus T$. ■

10.1.2 Error Correction

From error-reduction and linear-time encoding and decoding, we want to know if we can do error-correction. Let $R_2, R_4, R_8, \ldots$ be a sequence of error-reduction codes where R_k is $(\frac{1}{2}, \gamma)$-error-reducing. We combine a

bunch of these codes together to form a code, C_k. C_k has k info bits and $3k$ correction-bits. We break the message into several pieces and use different codes on the different pieces to give good correction parameters. Given (x, y) corresponding to (m, c) in the error-reducing codes we use (by induction on k) $C_{k/2}$ for y to get y'. Then we use R_{2k} on y, y' to get y''. This can be done in linear time. If R_k takes αk time and if $C_{k/2}$ takes $\beta k/2$ time, then the total time is $\alpha k + \beta k/2 + \alpha 2k \leq \beta k$, if β is sufficiently larger than α.

The correction algorithm is done as follows. Use the error-reduction algorithm on $(y \circ y', y'')$ to get (y_1, y_2). Then error correct on (y_1, y_2) to get y_3. Finally, error reduce on (x, y_3) to get m. If R_i are $(\frac{1}{2}, \gamma)$ -error-reducing codes then C_k corrects up to $\gamma/4$ fraction of errors, in linear time.

Proof: $\Delta(x \circ y \circ y' \circ y'', m \circ c \circ c' \circ c'') \leq \gamma n/4 = \gamma k$.
$\Rightarrow \Delta(y'', c'') \leq \gamma k$ and $\Delta(y \circ y', c \circ c') \leq \gamma k$.
$\Rightarrow$ by the reduction property $\Delta(y_1 \circ y_2, c_1 \circ c_2) \leq \gamma 2k/4 = \gamma k/2$ as $\epsilon = \frac{1}{2}$.
$\Rightarrow C_{k/2}$ corrects to get $y_3 = c$
$\Rightarrow$ given $c, \Delta(x, m) \leq \gamma k$ error reducing to get $x' = m$. ■

References

[1] D. Spielman (1996) Linear-time encodable and decodable error-correcting codes. Codes and complexity. IEEE Trans. Inform. Theory **42. no. 6, part 1**, 1723–1731.