6.440 Essential Coding Theory Mar 13, 2013

Lecture 11

Lecturer: Madhu Sudan *Scribe: Chiheon Kim*

In this lecture, we will discuss how to decode concaternated codes. We start with a simple decoding algorithm and discuss the better algorithm of Forney. Using this algorithm, we can achieve the capacity on the binary symmetric channel of Shannon. Beforehand, we need to finish the discussion in the last lecture.

1 Review: Error-locating Pairs Algorithm

Last time we discussed how to decode Reed-Solomon codes. More explicitly, we wanted to find "message" $m(x)$ from the evaluation $(\beta_1, \cdots, \beta_n)$ on n distinct points $(\alpha_1, \cdots, \alpha_n)$. For, we first find a pair of polynomials $(N, E) \neq (0, 0)$ with $\deg E \leq t$ and $\deg N < k + t$ such that $N(\alpha_i) = E(\alpha_i)\beta_i$ for all $i = 1, \cdots, n$. If N/E is again a polynomial, we return $N/E = m(x)$. By this algorithm, we can correct s erasures and t errors provided that $2t + s < n - k$ in polynomial time.

We may abstactize this scheme: think that N and E are from other codes under some conditions. Define $E * C = \{(e_1c_1, \cdots, e_nc_n) : e \in E, c \in C\}$, and assume $E * C \subset N$. How can we decode? For given $\beta = (\beta_1, \cdots, \beta_n)$, consider the following algorithm.

1. As we did for Reed-Solomon codes, find $e \in E$ and $w \in N$ such that $(e, w) \neq (0, 0)$ and $e_i\beta_i = w_i$ for all i. $e_i = 0$ if β_i is corrupted.
2. Set β_i to be "?" if $e_i = 0$.
3. Decode the modified β by erasure-decoding.

To correct t errors, we need the followings:

1. For the existence of nonzero $e \in E$ such that $e_i = 0$ on at most t places, we need $\dim(E) > t$.
2. For any (e, w) satisfying the first step, if $c \in C$ is the original codeword, then $e * c = w$. Note that $\Delta(e * c, e * \beta) \leq \Delta(c, \beta) \leq t$. So, it is enough to have $\Delta(N) > t$ to guarantee $e * c = w = e * \beta$.
3. To use erasure-decoding, we need to assure that β satisfying $e * \beta = w$ is unique. Such β could disagree at the places that $e_i = 0$, so at most $n - \Delta(E)$ places. Hence, we can guarantee the success of the third step if $\Delta(C) > n - \Delta(E)$.

Note that those three are simple linear algebraic conditions. The only non-trivial condition to achieve is $E * C \subset N$. If we randomly select N and E, then $\dim(N)$ is roughly $\dim(E)\dim(C)$. We need $\dim(E)$ and $\Delta(N)$ both to be large, but $\Delta(N)$ is small if its dimension is large. In the case of Reed-Solomon codes, we had $\dim(N) = \dim(E) + \dim(C)$. The only known other examples are algebraic geometry (AG) codes.

2 Decoding Concatenated Codes

2.1 Simple decoder

Recall the definition of concatenated codes. From "outer code" $[N, K, D]_Q$ and "inner code" $[n, k, d]_q$ with $Q = q^k$, Concatenating two leads us to a new code $[Nn, Kk, Dd]_q$. To be precise, we encode message $m = (m_1, m_2, \cdots, m_K) \in \mathbb{F}_Q^K$ as follows:

1. Encode m with outer code and obtain $(x_1, x_2, \cdots, x_N) \in \mathbb{F}_Q^N$.
2. Encode each alphabet $x_i \in \mathbb{F}_Q$ (as a string in $\mathbb{F}_q^k$) by inner code to have $y_i = (y_{i1}, \cdots, y_{in})$ where $y_{ij} \in \mathbb{F}_q$.
3. Return $(y_{11}, \cdots, y_{1n}, \cdots, y_{N1}, \cdots, y_{Nn})$.

If we receive $(r_{11}, \cdots, r_{Nn})$, how can we decode it? The simplest idea is by brute force to find $z_i \in \mathbb{F}_Q$ such that $\Delta(E_{inner}(z_i), r_i)$ is is minimum, and decode $(z_1, \cdots, z_N) \in \mathbb{F}_Q^N$. If the outer code was Reed-Solomon, we can decode it in $\text{poly}(N)$, and inner code will take $\text{poly}(N, Q)$ to find z_i.

How many errors would it correct? In each block, we can correct $< \frac{d-1}{2}$ errors. Also, we can correct $< \frac{D-1}{2}$ blocks which contains more than $\frac{d-1}{2}$ errors. Hence, this algorithm can correct $< \frac{(D-1)(d-1)}{4}$ errors. This tells us the concatenated code is $[Nn, Kk, Dd/2]_q$. In 1966, Forney gave a better algorithm for decoding concatenated codes which can correct $Dd/2$ errors. He used this idea to achieve Shannon capacity in constructive way.

2.2 Forney's constructive Shannon bound

Binary symmetric channel $BSC(p)$ with $0 < p < 1$ is the channel that flips each bit with probability p. We have seen that if $k = 1 - H(p) - \epsilon$, then there exist an encoder $E : \{0,1\}^k \to \{0,1\}^n$ and a decoder $D : \{0,1\}^n \to \{0,1\}^n$ such that $\Pr_\eta[D(E(m) + \eta) \neq m] \leq \exp(-\epsilon^{10} n)$ for every message m. In this context, E and D are chosen completely random. Can we construct such E and D in polynomial time? Before we see what Forney did, let's see how we can get polynomial bound, namely n^{-100}. First, find a good code with message length $100\epsilon^{-10} \log n$ by brute force. We can do it in polynomial time. Then divide the message into blocks with length $100\epsilon^{-10} \log n$, and encode each block separately. Decoding will fail with probability at most n^{-100}, so the entire decoding will fail with probability at most n^{-99} by union bound.

How can we even reduce n^{-100} to an exponential bound? Forney used concatenated codes to resolve this problem. Let the outer code be $E_{outer} : \mathbb{F}_Q^{(1-\epsilon)N} \to \mathbb{F}_Q^N$ which is Reed-Solomon (hence $Q = N$), and inner code be $E_{inner} : \mathbb{F}_2^k \to \mathbb{F}_2^n$ with $k = (1 - H(p) - \epsilon)n$ and $Q = 2^k$. Then,

- For each $i = 1, \cdots, N$, $\Pr[i\text{th block is decoded incorrectly}] \leq \exp(-\epsilon n)$.
- The probability that the number of incorrectly decoded block exceeds $\epsilon N/2$ is less than $2^{-\epsilon N/2}$. (Chernoff)

It perfectly runs in polynomial time if we only focus on N. How about ϵ? We need $k \geq 1/\epsilon^2$, so running time would be $\geq 2^{1/\epsilon^2}$, which seems bad. Also, he wanted to replace $-\epsilon N/2$ by $-\epsilon N$, which is analogous to find a decoder for concatenated codes which can correct $Dd/2$ errors instead of $Dd/4$.

2.3 Generalized minimum distance decoding

The main idea of Forney was that treating erasures is easier than treating errors. Recall that we can correct s erasures and t errors if $s+2t < D$ for Reed-Solomon codes. Here, we will use the same notations as in section 2.1, and assume that the outer code is a Reed-Solomon code.

Let $\hat{e}_i$ be the actual number of errors in block i, i.e., $\Delta(y_i, r_i)$, and let e_i be the number of errors "seen" in block i, i.e., $\Delta(r_i, E_{inner}(z_i))$. We want to decode correctly if $\sum \hat{e}_i < \frac{Dd}{2}$. For, we just erase each block with probability $\min\{1, \frac{e_i}{d/2}\}$, and use erasure-decoding for outer code. If the number of erased blocks plus twice the number of error blocks (which are not erased but wrong) is at most D in expectation, we can correct the code. The following claim will guarantee it.

Claim 1 *Set $A_i = 1$ if block i is erased, and $B_i = 1$ if block i is not erased but it is wrong ($x_i \neq z_i$). Then,* $\mathrm{Exp}[A_i + 2B_i] \leq \frac{2\hat{e}_i}{d}$.

Proof We have two cases: $z_i = x_i$ or $z_i \neq x_i$. For the first case, note that $e_i = \hat{e}_i$ and $B_i = 0$. Hence $\mathrm{Exp}[A_i + 2B_i] = \Pr[A_i = 1] = \frac{2e_i}{d} = \frac{2\hat{e}_i}{d}$. For the latter case, note that $\hat{e}_i \geq d - e_i$ since $\Delta(y_i, E_inner(z_i)) \geq d$. We have $\Pr[A_i = 1] = \frac{2e_i}{d}$ and $\Pr[B_i = 1] = 1 - \Pr[A_i = 1] = 1 - \frac{2e_i}{d}$, so $\mathrm{Exp}[A_i + 2B_i] = 2 - \frac{2e_i}{d} \leq 2 - \frac{2(d-\hat{e}_i)}{d} = \frac{2\hat{e}_i}{d}$. ■

By the claim, we get $\mathrm{Exp}[\sum_i (A_i + 2B_i)] \leq \frac{2}{d} \sum_i \hat{e}_i < D$, as desired.

This algorithm is not deterministic. How can we derandomize it? Actually we didn't use any independence between those A_i's. Hence, we may use only one random threshold $x \in [0, 1]$ to determine each block to be erased ($x \leq 2e_i/d$) or not ($x > 2e_i/d$). Further, we can try every possible threshold to get deterministic algorithm.