Introduction to Error Correcting Codes

Amir Shpilka
Arazim ©

January 13, 2016

In this lesson

1. Complexity of correction in a general linear code.
2. A cryptographic scheme based on the difficulty of decoding a linear code.
3. Secret sharing.
4. Recap of course.

1 The nearest codeword problem

Given a matrix $G \in M_{k\times n}$ over $\mathbb{F}_2$ and a word $y \in \mathbb{F}_2^n$ and a parameter δ, decide if there is a word with a distance $\leq \delta n$ from y.

Theorem 1. *The nearest codeword problem is in* **NPC***.*

Proof. We will introduce the following problem, which is **NPC**and show a reduction from our problem.

Problem 1 (Max-cut). Given a graph $\Gamma = (V, E)$ and a parameter s, decide whether there is a cut in the graph such that $|S, S^c|$

In our case, let $\Gamma = (V, E)$ be a graph. We will define the matrix $G \in M_{|V|\times|E|}$ for all $(u, v) \in E$ we will set $G_{e,v} = G_{e,u} = 1$.
There exists a cut with size $\geq S$
$\Leftrightarrow$ there exists a message $\mathbb{1}_A$ with $wt(G \cdot \mathbb{1}_A)$
$\Leftrightarrow$ the distance of the code word $G \cdot \mathbb{1}_A$ from the vector $\bar{1}$ is at most $|E| - s$. Thus $\delta = \frac{|E|-s}{|E|}$ □

Theorem 2. *There exists a constant $\gamma > 1$ such that the following problem is* **NPC***. Given a graph $\Gamma = (V, E)$ and a parameter s, decide whether there is a cut with a size $\leq s$ or there exists a cut with a size $\geq \gamma \cdot s$*

Corollary 1. *There is a constant for which it is hard to approximate the nearest codeword problem.*

Problem 2 (Approximating NCP). For any constant $\eta > 0$ the following problem is **NPC**. Given a generating matrix $G \in \mathbb{F}_2^{n\times k}$, a parameter δ and a codeword $y \in \mathbb{F}_2^n$, decide whether there is a codeword whos distance from y is at most δn, or every codeword is at least $\eta \cdot \delta \cdot n$ from y.

Proof. For every generating matrix G, parameter η and vector $\bar{y}$ we will define a new matrix G', vector y' such that if we can solve APX-NCP G', y' and a parameter η^2 then it is possible to solve APX-NCP for G, y and a parameter η and this is sufficient.
Construction: instead of writing G' we will describe a codeword. For all $n + 1$ original codewrds

$b, c_1, \ldots, c_n \in C$ we will define a new code words with a length n^2. every codeword will be a matrix with a size $n \times n$ and the word corresponding to $(b, c_1, \ldots, c_n)$ is

$$\begin{pmatrix} — & \text{b}— \\ \vdots & \vdots \quad \vdots \\ — & \text{b}—_{+} \end{pmatrix} \begin{pmatrix} | & \cdots & | \\ c_1 & \cdots & c_n \\ | & \cdots & | \end{pmatrix}$$

It is obvious that the new code is linear. We will show that if C is the nearest codeword to y at a distance of t then in C', the nearest codeword y' is at a distance of t^2. For simplicity we will assume that $\bar{y} = \bar{1}$ and define $y' = (\bar{y})$ (matrix with only 1's).

Let $x \in C$ be the closest codeword to y. $x = (\overbrace{0, \ldots, 0}^{t}, 1, \ldots, 1)$.

$$b = x \qquad c_i = \begin{cases} x & x_i = 0 \\ 0 & x_i = 1 \end{cases}$$

And the codeword matching $(b, c_1, \ldots, c_n)$ is

$$\begin{pmatrix} 0|1 \end{pmatrix} + \begin{pmatrix} 0|0 \\ 1|0 \end{pmatrix} = \begin{pmatrix} 0|1 \\ 1|1 \end{pmatrix}$$

We have found a codeword in C' whos distance from y' is exactly t^2. □

2 McEliece scheme

An encryption scheme with a public key. We will assume that C is a linear code with a generating matrix $G \in M_{n \times k}$ for which there is an efficient algorithm for t errors.
Key creation:

1. We will randomize an invertible matrix $k \times k$ A.
2. We will randomize a permutation matrix $\pi : [n] \to [n]$ and let P be a matrix that represents it.

- Secret key: A, G, P.
- Public key: $G' = P \cdot G \cdot A$ (we assume that G is known).
- Encryption: Given a message $x \in \{0, 1\}^k$, Alice will randomize a vector $z \in \{0, 1\}^n$ such that z has t 1's.
- The encrypted text: $G' \cdot x + z$.
- Given a y, and using the secret key we will calculate P^{-1} and use it as follows:

  $$P^{-1} \cdot y = P^{-1}G' \cdot x + P^{-1}z = P^{-1} \cdot PGAx + P^{-1}z = G \cdot A\bar{x} + z'$$

  where z' has the same weight t. We will run the code correcting algorithm and arrive at $A \cdot \bar{x}$ and after multiplying by A^{-1} we arrive at the original x.

3 Secret sharing

There is a secret s and a parameter $n \geq t$, We want to divide s to n people such that each person will recieve a b_i.
Requirements:

1. Every $\geq t$ people will be able to find s from their parts

2' No $t-1$ people will be able to find s.

2. No $t-1$ people will be able to glean any information from the code.

Construction:
We will assume that $S \in \mathbb{F}, |\mathbb{F}| > n$ and we will randomize a polynomail $f(x)$ with a degree of $t-1$ over $\mathbb{F}$ for which $f(0) = s$. We will divide the codes by choosing different and nonzero $\beta_1, \ldots, \beta_n \in \mathbb{F}$

$$f(x) = \sum_{i=1}^{t-1} \alpha_i x^i + s \qquad b_i = f(\beta_i)$$

Claim 1. Every $t-1$ people can recover f and calculate s.

Claim 2. $t-1$ people have no information about s.

4 In this course

- Shannon bound.
- Classic codes: Hamming, Adamard, RS, RM.
- Bounds : Hamming, Singleton, Plotkin, GV.
- MDS codes (RS for example).
- Operations: Adding a prity bit, puncturing, multiplying and composition.
- Constructions (Justesen codes): using the Wozencraft example. (Note: we can approach the Shannon bound as much as we would like).
- Algorithms: RS W-B, RM, composition.
- List-decoding: Algorithm for RS, local RS (reduction from worst-case, average-case, hardness).
- Bound on list-decoding: Johnson bound, bound on RS.
- Elias Bassalygo bound.
- Expanding graphs and codes.
- A construction of linear time decoding and encoding.
- Effcient codes which meet the Shannon bound.
- Complexity of decrypting a "random" code.
- An encryption scheme.
- A secret sharing scheme.