Polys. (& factoring) in Coding theory

Basic problem: Alice wants to send Bob N bits through a channel having t bit-errors.

How to communicate correctly in minimum bits?

Trivial soln: Alice sends Bob a message with enough redundancy.

(eg. $N \cdot (2t+1)$ bits suffice.

Encode each bit with a $(2t+1)$-string block of repetition.

Decode each block by taking the majority vote.)

Clever algebraic soln:

Reed & Solomon (1960) gave a code requiring $O(N \cdot \lg N)$ bits, that corrects around $N/2$ bit-errors.

- RS codes are very widely used in:
  (1) mass storage systems,
  eg. CD, DVD, distributed online storage.

  (2) bar codes

  (3) deep space & satellite communications.

Reed-Solomon Code

- View the message as a polynomial over a finite field.
  Send the evaluations over the channel.

Encoding: (1) Break the N-bit message m into k blocks each of size b-bits.
View these blocks as elements

$d_0, d_1, \ldots, d_{k-1}$ in the field $\mathbb{F}_{2^b}$.

(2) Define $P(x) := d_0 + d_1 x + \ldots + d_{k-1} x^{k-1}$
$\in \mathbb{F}_{2^b}[x]$.

(3) Pick n distinct points $e_0, \ldots, e_{n-1} \in \mathbb{F}_{2^b}$.
Send the code $(c_0, c_1, \ldots, c_{n-1}) :=$
$(P(e_0), P(e_1), \ldots, P(e_{n-1}))$.

▷ The encoding is a linear map from
$\{0,1\}^N = (\mathbb{F}_{2^b})^k$ to $(\mathbb{F}_{2^b})^n = \{0,1\}^{bn}$.

▷ It can be computed in $\tilde{O}(nb)$ time.

- The code $\bar{c} := (c_0, \ldots, c_{n-1})$ gets transmitted over the <u>erroneous</u> channel.

- If there are <u>no</u> errors, then Bob can interpolate P from $\bar{c}$, assuming $2^b \geq n \geq k$.

Decoding RS

- How does Bob decode m from a corrupted version $\bar{c}'$ of $\bar{c}$?

- Let there be t errors: Say, the values $P(e_{i_1}), \dots, P(e_{i_t})$ are wrong.

- (Peterson 1960) The main idea is to consider the error locator polynomial $Q(x) := \prod_{j \in [t]} (x - e_{i_j})$.

$$\Rightarrow (c_j - c'_j) \cdot Q(e_j) = 0 \quad , \forall 0 \le j \le n-1.$$

$$\Rightarrow P(e_j) \cdot Q(e_j) = c'_j \cdot Q(e_j)$$

$$\Rightarrow R(e_j) = c'_j \cdot Q(e_j)$$

where, $R(x) := P \cdot Q \in \mathbb{F}_{2^t}[x]$.

- We do not know Q & R.

- But, we do know their degree bounds: $\deg R = k-1+t$ &
$$\deg Q = t.$$

$\Rightarrow$ The #unknowns is $(k-1+t)+1+t$
$$= k+2t.$$

<u>Claim</u>: Every solution R, Q of the linear system:
$$R(e_j) = c_j' \cdot Q(e_j), \ \forall 0 \le j \le n-1,$$
will satisfy $Q | R$ if
$$n \geqslant k+2t.$$

$\triangleright$ The original message is $P(x) := R/Q$.

<u>Correctness Pf</u>:

- Let $2^b \geqslant n \geqslant k+2t$.
- The linear system has at least one solution, namely $Q =$ error-locator & $R = P \cdot Q$.

- Let Q', R' be some other solution.
- From the linear system we know that the polynomial $\Delta(x) := R' - P.Q'$ vanishes on at least $(n-t)$ points in $\{e_0, e_1, \dots, e_{n-1}\}$.

- On the other hand, $\deg \Delta \leq k-1+t < n-t$.

$\Rightarrow$ The number of distinct roots of Δ is $> \deg \Delta$.

$$\Rightarrow \quad \Delta = 0$$

$$\Rightarrow \quad R'/Q' = P(x).$$

$\square$

- Time complexity claims:

1) The linear system is special & can be solved in $\tilde{O}(nb)$-time.
2) Overall, time complexity is $\tilde{O}(nb)$.

- One could find $R'(x)/Q'(x)$ by interpolation, avoiding division.

Distance

- Let us fix the parameters:
  $$b = \lg N, \quad k = \frac{N}{\lg N}, \quad n = N.$$

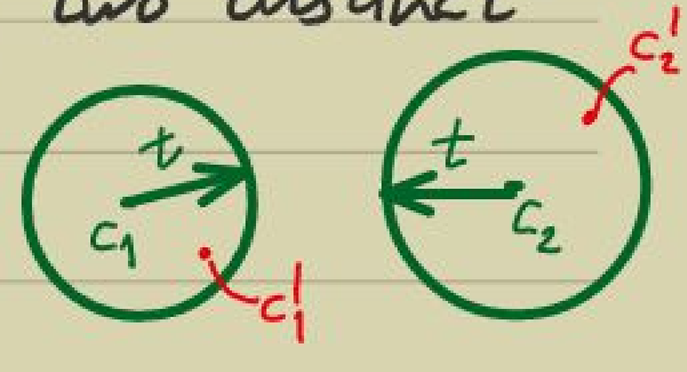
- Then, RS decoder works when
  $$t \leq \frac{n-k}{2} = \frac{N}{2}\cdot\left(1 - \frac{1}{\lg N}\right).$$

▷ RS code is of <u>length</u> $N \cdot \lg N$ & corrects up to $\frac{N}{2}\cdot\left(1-\frac{1}{\lg N}\right)$ <u>errors</u>.

around 50% correction in terms of field elts.

- $(2t+1)$ is called the <u>distance</u> of the code. (in this case, <u>non-binary</u> alphabet)

- Intuitively, it is the <u>minimum Hamming distance</u> between any two distinct codewords!



Crossing the 50% barrier

- When the error bound $t \geq N/2$, then there are many messages corresponding to a corrupted codeword.

- Could we find all of them?

- (Madhu Sudan, 1995) found an efficient way to find them —

List Decoding.

- Consider the scenario:

$$d_0, \ldots, d_{k-1} \xmapsto{RS} \underset{P(e_0)}{\overset{\shortparallel}{c_0}}, \ldots, \underset{P(e_{n-1})}{\overset{\shortparallel}{c_{n-1}}} \xmapsto[\left(\substack{\geq T \text{ of the} \\ c_i\text{'s are correct}}\right)]{\text{channel}} c_0', \ldots, c_{n-1}' .$$

- Now consider a bivariate "error locator" polynomial $Q(x, y)$ of degree D_x & D_y s.t.

$$Q(e_j, c_j') = 0, \quad \forall\, 0 \le j \le n-1.$$

[If $(1+D_x)(1+D_y) \geq n$ then such a nonzero Q exists, and can be computed by linear algebra.]

- Consider $R(x) := Q(x, P(x))$.
  It has $\deg \le D_x + (k-1)\cdot D_y$.
  We know that $R(e_j) = 0$ for T many j's in $[0, n-1]$.

$\Rightarrow$ If $T > D_x + (k-1)D_y$, then $R(x) = 0$, hence, $(y - P(x)) \mid Q(x,y)$.

Lemma: If $n < (1+D_x)(1+D_y)$ & $D_x + (k-1)D_y < T$, then a curve Q fitting $\{(e_j, c_j') \mid j\}$ has $(y - P(x))$ as a factor.

- Finally, the decoding algorithm is:

1) Fix the parameters:
$D_x = \sqrt{nk}$, $D_y = \sqrt{n/k}$ & $T = 2\sqrt{nk}$.

2) Compute $Q(x,y)$ with degree D_x, D_y s.t. $\forall\, 0 \le j \le n-1$: $Q(e_j, c_j') = 0$.

3) Factor $Q(x,y)$ & collect its factors of the form $y - f(x)$ with $\deg f \le k-1$.
[They can be at most D_y many.]

4) <u>Output</u> the list of such $\{f\}$.

▷ This list-decoding algorithm is in randomized poly-time.
It works up to $(n - 2\sqrt{nk})$ many bit errors! Eg. For $n = k\lg^2 k$, we only need $2k\lg k$ correct values!

– Later, we'll learn bivariate poly. factoring.

- In the decoding of RS codes we needed two <u>new</u> algebraic operations:
1) construction of a finite field, &
2) factoring a bivariate polynomial.

<u>Constructing the field $\mathbb{F}_q$</u>.

- Let $q = p^{\ell}$. Then, we want to find an irreducible polynomial over $\mathbb{F}_p$ of deg ℓ.

- We will show that a random choice works!

- Let $\pi(\ell)$ denote the number of irreducible polynomials in $\mathbb{F}_p[X]$ of degree ℓ.

- Recall that the polynomial $X^{p^\ell} - X$ has, as factors, <u>all</u> irreducible polynomials of degree $k | \ell$.

▷ Thus, $p^{\ell} = \sum_{k|\ell} k \cdot \pi(k)$.

- This identity leads to a "prime number thm" for polynomials.

<u>Theorem</u>: $\forall \ell \geqslant 1$, $\frac{p^\ell}{2\ell} \leqslant \pi(\ell) \leqslant \frac{p^\ell}{\ell}$ &

$$\pi(\ell) = p^\ell/\ell + O\left(p^{\ell/2}/\ell\right).$$

<u>Proof</u>: • From the previous identity, we deduce:

$$\ell \cdot \pi(\ell) = p^\ell - \sum_{\substack{k|\ell \\ k<\ell}} k \cdot \pi(k)$$

$$\geqslant p^\ell - \sum_{k|\ell, k<\ell} p^k$$ [$\because$ the above identity gives $k \cdot \pi(k) \leqslant p^k$.]

$$\geqslant p^\ell - \sum_{k=1}^{\lfloor \ell/2 \rfloor} p^k \geqslant p^\ell - \frac{p}{p-1} \cdot (p^{\ell/2} - 1).$$

$$\Rightarrow \ell \cdot \pi(\ell) = p^\ell + O(p^{\ell/2}).$$

• Moreover, $\frac{p}{p-1} \cdot (p^{\ell/2} - 1) \leqslant \frac{1}{2} \cdot p^\ell$, $\forall p \geqslant 2, \ell \geqslant 1$.

$$\Rightarrow \ell \cdot \pi(\ell) \geqslant p^\ell/2 \quad (\& \leqslant p^\ell).$$

$\square$

- Thus, if we pick a random degree b polynomial in $\mathbb{F}_p[X]$, then it will be <u>irreducible</u> with probability $\geqslant 1/2b$.

- On repeating this experiment $2b$ times, the probability of success is $\geqslant 1-\left(1-\frac{1}{2b}\right)^{2b}$

$$= 1-\left(1-2b\cdot\frac{1}{2b}+\frac{2b\cdot(2b-1)}{2}\cdot\frac{1}{4b^2}-\cdots\right) > \frac{1}{2}.$$