Today
- Unique decoding
  RS codes

CSS.318.1
Coding Theory
Lecture 10 (2022-9-26)
Instructor: Prahladh Harsha.

Algorithmic Issues:

$C \subseteq \Sigma^n$ $\qquad$ C - $(n,k,d)_q$-code.

1. Encoding
2. Detecting Errors
3. Correcting Erasures
4. Correcting Errors

(1)–(3): Easy if code is <u>linear</u>.

1, 2, 3 - easy if code is linear & we have access to G & H.

Encoding is efficient if

$$\underbrace{C : \Sigma^k \to \Sigma^n \text{ is efficient.}}_{\text{explicitness of a code.}}$$

(4) Problem. Given a code $\mathcal{C}$ (how? via generator matrix or Encoding map) & $x \in \Sigma^n$ find $c \in \mathcal{C}$, s.t $\Delta(x, c)$ is minimized.

As stated above, the problem is NP-hard even for specific codes (such as RS codes)

Shannon: Can we solve it for most x (that arise out of a channel)?

Hamming: Promise(t): $\exists\, c \in \mathcal{C}$, $\Delta(x, c) \leq t$.

Our approach:- Hamming's way out
Find largest t for which we can solve the problem.

Unique-Decoding
Combinatorially: If $t < d/2$, there exists at most one $c \in \mathcal{C}$, s.t $\Delta(x, c) \leq t$.

Algorithmically ??

Reed-Solomon Code: History
- Peterson 1960
- Gorenstein-Zierler

$\Big\}\ O(n^3)$

Unique Decoding

- Berlekamp
  Massey } $O(n^2)$

(now, nearly linear $O(n \operatorname{polylog} n)$ time algorithms)

- Welch-Berlekamp '86 $O(nt)$
- Gemmell-Sudan '92 (reinterpretation of WB algorithm).

List-decoding ($t > d/2$)

$t > d/2$

Sudan '95

Guruswami-Sudan '98 (t - Johnson Radius)

Today: Gemmell-Sudan style of unique decoding RS codes

Problem

Input: $\mathbb{F}$ - finite field, $|\mathbb{F}| = q$.

$S = \{\alpha_1, \ldots \alpha_n\}$, $|S| = n$. $S \subseteq \mathbb{F}$

k - degree parameter.

t - bd on errors

$\bar{\beta} = (\beta_1, \ldots \beta_n) \in \mathbb{F}^S$ (received word).

Output: Find all polynomials $p \in \mathbb{F}_{<k}[X]$ s.t

$\#\{i \in [n] \mid p(\alpha_i) \neq \beta_i\} \leq t$

If $2t < d$, there is at most one such poly. and let p be the unique poly (if one exists)

$$\mathrm{Err} = \{ i \in [n] \mid p(\alpha_i) \neq \beta_i \}$$ (Don't know Err).

Error-locator polynomial

polynomial whose zeroes are the errors

$$\hat{E}(x) = \prod_{i \in \mathrm{Err}} (x - \alpha_i)$$ (Don't know $\hat{E}$)

Properties of $\hat{E}$:

(1) $\hat{E} \neq 0$, $\deg \hat{E} \leq t$.

(2) $\forall i \in [n]$, $p(\alpha_i) \cdot \hat{E}(\alpha_i) = \beta_i \cdot \hat{E}(\alpha_i)$

$$\hat{N}(x) \triangleq p(x) \cdot \hat{E}(x)$$

(a) $\deg \hat{N} \leq t + k - 1$

(b). $\forall i \in [n]$, $\hat{N}(\alpha_i) = \beta_i \hat{E}(\alpha_i)$

Qn: Can we find poly E of $\deg \leq t$ s.t $\exists$ $\hat{N}$ of $\deg \leq t+k-1$ that satisfies (a) & (b).

Issues: (1) How do we find such an E?

(2) Why is such an E useful?

Welch-Berlekamp Algorithm:

Step 1:

Find non-zero (N, E) - pair of polynomials s.t.

- $\deg(E) \leq t$
- $\deg(N) \leq t + k - 1$
- $\forall i \in [n]$, $N(\alpha_i) = \beta_i E(\alpha_i)$

Step 2: Output N/E (if it is a polynomial).

BW algorithm: Efficient

- Step 1 - linear system
  - in #var $2t + k + 1$
  - #cons n
- Step 2 · division.

Claim 1: If $\exists$ poly. $\#\{i \in [n] / p(\alpha_i) \neq \beta_i\} \leq t$ then there is a non-zero soln.

Claim 2: Let (N_1, E_1) & (N_2, E_2) be any two non-zero pairs of solns to Step 1 then
$(t < \frac{n-k+1}{2})$

$$\frac{N_1}{E_1} = \frac{N_2}{E_2}.$$

Note: Claims 1-2 $\Rightarrow$ Correctness of BW algorithm.

Proof of Claim 1: $(\hat{N}, \hat{E})$ is a non-zero soln. $\boxtimes$

Proof of Claim 2:

Need to show

$$N_1 E_2 \equiv N_2 E_1$$

Appeal to degree mantra.

$$\deg(N_1 E_2),\ \deg(N_2 E_1) \leq t + t + k - 1$$
$$= 2t + k - 1$$

$$\forall i \in [n] \quad N_1(\alpha_i) E_2(\alpha_i) = \beta_i E_1(\alpha_i) E_2(\alpha_i)$$
$$= E_1(\alpha_i) N_2(\alpha_i)$$

If $n > 2t + k - 1$, then $N_1 E_2 \equiv N_2 E_1$ $\boxtimes$

Obs:

Claim 2 :$\Rightarrow$ any E obtained in Step 1 is a multiple of $\hat{E}$

E satisfies $\quad N(\alpha_i) = \beta_i E(\alpha_i)$

$$p(\alpha_i) \cdot E(\alpha_i) = \beta_i E(\alpha_i)$$

$$p(\alpha_i) \neq \beta_i \Rightarrow E(\alpha_i) = 0$$
$$\hat{E}(\alpha_i) = 0 \Rightarrow E(\alpha_i) = 0$$

Hence $\hat{E} \mid E$.

An alternate way to solve the below question

Qn: Can we find poly E of deg $\leq t$ s.t. $\exists$ $\hat{N}$ of deg $\leq t+k-1$ that satisfies (a) & (b).

Alternate Approach:

$(E \odot \beta)$ - pointwise product (Hadamard product)

$\in RS_{\mathbb{F}}[S, t+k]$

Equivalently $E \odot \beta \perp RS_{\mathbb{F}}[S, t+k]^{\perp}$ ... $(*)$

For simplicity work with $S = \mathbb{F}_q^*$; $n = q-1$.

In this case we know

$$RS_{\mathbb{F}}[\mathbb{F}^*, k]^{\perp} = \text{Span}\{ \text{Eval}_S(x^{\ell}) \mid 1 \leq \ell \leq n-k \}$$

$(*)$ can be written as.

$$\forall\ 1 \leq \ell \leq n-t-k, \quad \sum_{i=0}^{n-1} \left(E(\alpha^i)\beta_i\right) \cdot (\alpha^i)^{\ell} = 0$$

$$E(x) = \sum_{j=0}^{t} E_j x^j$$

$$\sum_{i=0}^{n-1} \sum_{j=0}^{t} E_j \alpha^{ij} \beta_i \alpha^{il} = 0, \quad \forall\, 1 \le l \le n-t-k$$

$$\sum_{j=0}^{t} E_j \underbrace{\sum_{i=0}^{n-1} \beta_i \alpha^{(j+l)i}}_{\shortparallel} = 0, \quad \forall\, 1 \le l \le n-t-k \quad \ldots (\ast\ast)$$

$$S_l := \langle \bar{\beta}, \mathrm{Eval}_S(x^{j+l}) \rangle$$

β - purportedly eval of deg k poly

$\mathrm{Eval}_S(x^{l+j})$: $1 \le l+j \le n-k$

Observe,

$$x^l \rightarrow \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{q-2} \\ 1 & \alpha^2 & \alpha^4 & \cdots & \alpha \\ & & \vdots & & \\ \cdots & & \alpha^{il} & & \\ & & & & \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n-1} \end{bmatrix} = \begin{bmatrix} S_0 \\ S_2 \\ \vdots \\ S_{n-k} \end{bmatrix}$$

$\swarrow \alpha^{i}$

Rewriting $(\ast\ast)$

$$\sum_{j=0}^{t} E_j S_{l+j} = 0 \;; \quad 1 \le l \le n-t-k \quad \ldots (\ast\ast\ast)$$

Step 1: Compute Syndrome $(S_1 \ldots S_{n-k})$

Step 2: Solve E that satisfies $(\ast\ast\ast)$

Step 3: Given E, find the err &
use erasure decoding.

Peterson, BM, subsequent improvements are just implementations of above idea.