ST05 Lecture 25

Note Title 5/16/2006

Today + Next Lecture

- Computational Perspectives on Source / Channel Coding.

Today: The Problem of Channel Coding.

Ex. BSC (p).

Source $\to$ Encode $\to$ BSC $\to$ Decoder $\to$ Receiver

Given $m \in \{0,1\}^k$

$E : m \longmapsto x = E(m) \in \{0,1\}^n$

Given $y \in \{0,1\}^n$

$D : y \longmapsto \hat{m} \in \{0,1\}^k$

Shannon: Fix $R = (1 - H(p) - \epsilon)$

As $n \to \infty$, $\exists$ E, D s.t.

$$\Pr[D(y) \neq m] \to 0$$

$m \to x \to y \to \hat{m}$

Problem: E, D exist.

Are they efficient to compute

e.g. if $k = 100$ bits

$n = 200$ bits

running time of D may be 2^{100}!

" E may be $2^{2^{100}}$!

Can we do better?

<u>Goal:</u> E, D must run in time poly(n).

Acheived in 1966 by [Forney]. How?

<u>Encoding</u>

REED - SOLOMON

$+$

CONCATENATION

<u>Decoding</u>

Brute force

$+$

Algebra.

REED-SOLOMON Codes

Imagine we are trying to Correct errors on a q-ary channel ; q = prime.

<u>Idea</u>: view alphabet $\Sigma = \{0, \ldots q-1\}$ as a "field" of addition / multiplication modulo q.

Polynomials over field $\mathbb{F}$

$$p \in \mathbb{F}[x] \; : \quad \langle c_0 \ldots c_{k-1} \rangle = \sum c_i x^i$$

Evaluation: p at α = $\sum c_i \alpha^i$ $\quad (Ev_{p} : \mathbb{F} \to \mathbb{F})$

Nice properties

degree l polynomial $p(x)$ has at most l roots in $\mathbb{F}$.

RS encoding

Message = k elements of $\mathbb{F}$

En n elements of $\mathbb{F}$

How? Fix $\alpha_1 \ldots \alpha_n$ distinct in $\mathbb{F}$

$$c_0 \ldots c_{k-1} \rightarrow p \in \mathbb{F}[x] \rightarrow \langle p(\alpha_1) \ldots p(\alpha_n) \rangle$$

Properties

$$\begin{array}{ccccc} c_0 \ldots c_{k-1} & \rightarrow & p & \Rightarrow & (p-p') \neq 0 \\ \neq & & \neq & & \Downarrow \\ c_0' \ldots c_{k-1}' & \rightarrow & p' & & \end{array}$$

$(p-p')(\alpha_i) = 0$ for at most $k-1$ choices of i

$\Rightarrow$ differ in at least $n-(k-1)$ places!

Immediate Consequences:

- Can correct upto $n-(k-1)$ erasures in poly time. How?

$p(\alpha_1) \; \ldots \ldots \qquad p(\alpha_n)$

$\downarrow$

$\langle \; ? \quad p(\alpha_2) \quad p(\alpha_3) \quad ? \quad ? \quad p(\alpha_6) \ldots \; p(\alpha_n) \rangle$

To find $C_0 \ldots C_{k-1}$ solve lin system

$$\begin{bmatrix} 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{k-1} \\ 1 & \alpha_3 & \alpha_3^2 & \cdots & \alpha_3^{k-1} \\ & & \vdots & & \\ 1 & \alpha_n & \alpha_n^2 & \cdots & \alpha_n^{k-1} \end{bmatrix} \begin{bmatrix} C_0 \\ \vdots \\ \\ C_{k-1} \end{bmatrix} = \begin{bmatrix} p(\alpha_2) \\ p(\alpha_3) \\ \vdots \\ p(\alpha_n) \end{bmatrix}$$

Solving errors ?

- Can correct $\frac{n-(k-1)-1}{2}$ errors in polytime [1960 - Peterson]

What does this have to do with the BSC ?

Forney

Let $q \approx 2^{\ell}$ (field element is ℓ bits).

take k bit message

$= \frac{k}{\ell}$ elements of $\mathbb{F}$

Encode as polynomial of degree $(1+\epsilon)\frac{k}{\ell}$.

yield $(1+\epsilon)\frac{k}{\ell}$ elements of $\mathbb{F}$

But now encode $\mathbb{F}$ elements as (say) 2ℓ bits by a good (Shannon) code from $\{0,1\}^{\ell} \to \{0,1\}^{2\ell}$.

- Distinct messages differ in at least $\epsilon \cdot \frac{\ell}{10}$ bits.

- To decode

Yields ... poly time enc. + dec. ...

Alas: No solution to 2^{100} problem.