Design of codes

1 Introduction

1.1 Design of codes

Recall our general problem: design a code:

- With high rate
- Which can detect many errors
- Which is easy to encode and decode

1.2 How do we design good codes?

Start with a good code, and modify it:

- cut it
- add a parity check bit
- Take a subset of the codewords
- Take the dual

2 EAN and ISBN

2.1 EAN

This uses a variant of the parity check code $c = (c_1, ..., c_{13})$ where:

$$c_{13} = -\sum_{i=0}^{5} (c_{2i+1} + 3c_{2i+2}) \mod 10$$

Ex: 5-045092-36551?

$$\begin{aligned} c_{13} &= -[5 + (3 \times 0) + 4 + (3 \times 5) + 0 + (3 \times 9) + 2 + (3 \times 3) \\ &+ 6 + (3 \times 5) + 5 + (3 \times 1)] \\ &= -(5 + 4 + 5 + 7 + 2 + 9 + 6 + 5 + 5 + 3) \\ &= -1 = 9 \end{aligned}$$

2.2 ISBN

This is another variant of the parity check code where

$$c_{10} = \sum_{i=1}^{9} ic_i \mod 11$$

Example: ISBN-10 number 0-262-06141-?

$$\begin{aligned} c_{10} &= [(1 \times 0) + (2 \times 2) + (3 \times 6) + (4 \times 2) + (5 \times 0) \\ &+ (6 \times 6) + (7 \times 1) + (8 \times 4) + (9 \times 1)] \\ &= 4 + 7 + 8 + 3 + 7 + 10 + 9 \\ &= 4 \end{aligned}$$

3 Introduction to algebraic codes

3.1 More structure

We can use polynomials for more complicated codes using sequences of digits

3.2 GF(4)

Let α be a root of $x^2 + x + 1$, i.e.

$$\alpha^2 + \alpha + 1 = 0, \quad \text{or equivalently,} \quad \alpha^2 = \alpha + 1$$

$+$	0	1	α	$\alpha+1$
0	0	1	α	$\alpha+1$
1	1	0	$\alpha+1$	α
α	α	0	$\alpha+1$	α
α	α	$\alpha+1$	0	1
$\alpha+1$	$\alpha+1$	α	1	0

$\times$	0	1	α	α^2
0	0	0	0	0
1	0	1	α	α^2
α	0	α	α^2	1
α^2	0	α^2	1	α

3.3 GF(8)

The construction can be extended for any $GF(2^m)$

E.g. GF(8). Let β be a root of $x^3 + x + 1$ i.e.

$$\beta^3 + \beta + 1 = 0$$

Then GF(8)=$\{0, 1, \beta, \beta^2, \beta^3 = \beta + 1, \beta^4 = \beta^2 + \beta, \beta^5 = \beta^2 + \beta + 1, \beta^6 = \beta^2 + 1\}$

3.4 Reed-Solomon codes

The code $RS(k, k)$ is the set of all evaluations of polynomials of degree at most $k-1$ over all nonzero elements of GF(q) where n=q-1

Let $q = 2^m$ and γ generate GF(q) i.e.

$$\mathrm{GF}(q) = \left\{0, 1, \gamma, \ldots, \gamma^{q-2}\right\}$$

For any polynomial c(x) with coefficients in GF(q), let

$$\mathbf{c} = \left(c(1), c(\gamma), \ldots, c\left(\gamma^{q-2}\right)\right) \in \mathrm{GF}(q)^n$$

Then

$$RS(n, k) = \{\mathbf{c} : \deg c(x) \leq k - 1\}$$

3.5 Generator matrix of Reed-Solomon Codes

E.g. for RS(7,2)

$$\mathbf{G}_{RS(7,2)} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \beta & \beta^2 & \beta^3 & \beta^4 & \beta^5 & \beta^6 \end{pmatrix}$$

E.g. for encoding $\left(\beta^2, \beta\right)$, i.e. $c(x) = \beta^2 + \beta x$

$$\begin{aligned} \mathbf{c} &= \left(c(1), c(\beta), c\left(\beta^2\right), c\left(\beta^3\right), c\left(\beta^4\right), c\left(\beta^5\right), c\left(\beta^6\right)\right) \\ &= \left(\beta^4, 0, \beta^5, \beta, \beta^3, 1, \beta^6\right) \end{aligned}$$

3.6 Bound on the minimum distance

The **Singleton bound**: if C is an (n, k, d_{min})-code, then

$$d_{min} \leqslant n - k + 1$$

Proof: look at the parity check matrix: the columns have size n-k

- At most n-k linearly independent columns
- Any set of n-k-1 columns is linearly dependent

3.7 Minimum distance of RS codes

For any two polynomials $c(x) \neq d(x)$ of degrees $\leqslant k - 1$

- $c(x) - d(x) \neq 0$ has degree $\leqslant k - 1$
- $c(x) - d(x)$ has at most $k - 1$ roots
- c and d agree on at most k-1 positions
- $d_H(c, d) \geqslant n - k + 1$

By the singleton bound, we obtain:

$$d_{min} = n - k + 1$$

3.8 RS Decoding

Due to their structure RS are easy to decode, but we won't go into that