

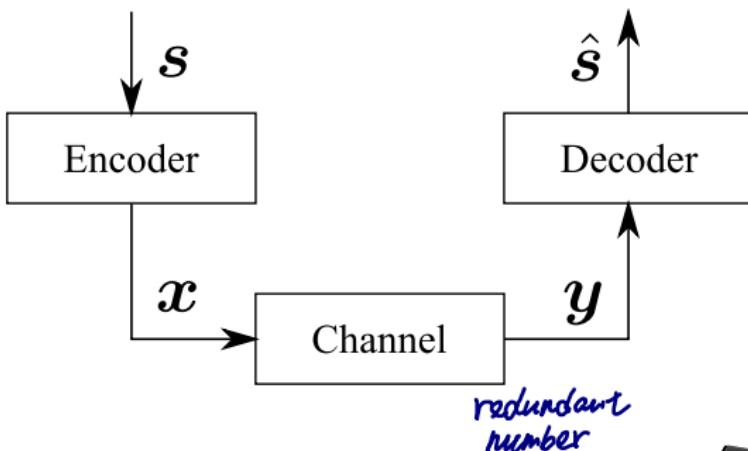
Section 3

Error Correcting Codes (ECC): Fundamentals

- ▶ Communication systems and channel models
- ▶ Definition and examples of ECCs
- ▶ Distance

For the contents relevant to distance, Lin & Xing's book, Chapter 2, should be helpful.

Communication Systems



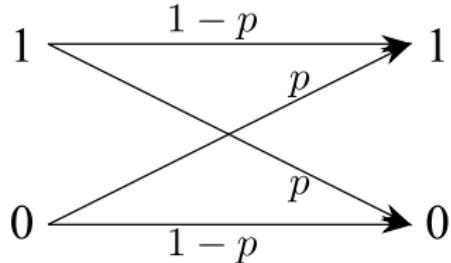
*redundant
number*

Abstract Channel Model: Binary Symmetric Channel (BSC)

Binary Symmetric Channel: a **memoryless** channel such that

$$\Pr(0 \text{ received} | 1 \text{ sent}) = \Pr(1 \text{ received} | 0 \text{ sent}) = p,$$

$$\Pr(1 \text{ received} | 1 \text{ sent}) = \Pr(0 \text{ received} | 0 \text{ sent}) = 1 - p.$$



p is called the **transition (cross-over) probability**.

Memoryless channel: A channel that satisfies

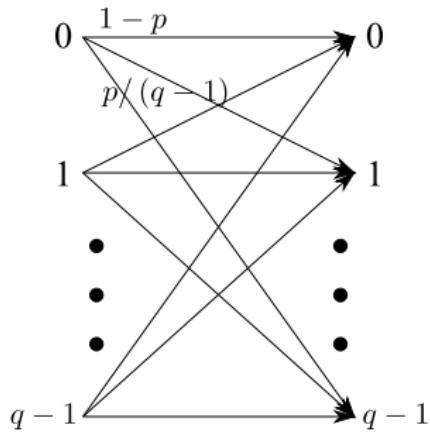
$$\Pr(y \text{ received} | x \text{ sent}) = \prod_{i=1}^n \Pr(y_i \text{ received} | x_i \text{ sent}).$$

The Memoryless q -ary Symmetric Channel

Define an **alphabet** set \mathbb{F}_q .

Both channel input x_i and channel output y_i are from \mathbb{F}_q .

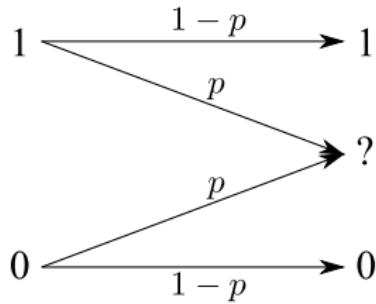
Crossover probability p :



$$\Pr(y_i|x_i) = \begin{cases} 1 - p & \text{if } y_i = x_i \\ p / (q - 1) & \text{if } y_i \neq x_i \end{cases}$$

The Memoryless Binary Erasure Channel (BEC)

Binary Erasure Channel:



- ▶ Internet traffic: a package got lost.
- ▶ Cloud storage: one copy of file messed up.

What is a Code?

Definition 3.1 (Code)

A code is a set \mathcal{C} containing (row) vectors of elements from \mathbb{F}_q .

An (n, M) block code: $\mathcal{C} \subset \mathbb{F}_q^n$ and $|\mathcal{C}| = M$.

A codeword: a vector in \mathcal{C} .

Codeword length: n

Code size: M

Dimension: $k = \log_q M$.

Rate: $r = k/n$.

Example 1:

$$\mathbb{F}_2 = \{0, 1\}. \mathcal{C} = \{0000, 1100, 1111\}.$$

$$n = 4. M = 3. k = \log_2 3 = 1.585. r = 0.3962.$$

Example 2:

$$\mathbb{F}_3 = \{0, 1, 2\}. \mathcal{C} = \{00000, 12121, 20202\}.$$

$$n = 5. M = 3. k = \log_3 3 = 1. r = 0.2.$$

Triple Repetition Code

Encoding

$$1 \rightarrow 111$$

$$0 \rightarrow 000$$

Decoding: majority voting

$$111, 110, 101, 011 \rightarrow 1$$

$$000, 001, 010, 100 \rightarrow 0$$

Error probability computation:

$$P(\hat{s} = 1 | s = 0)$$

$$= P(111|0) + P(110|0) + P(101|0) + P(011|0)$$

$$= p^3 + 3p^2(1-p)$$

$$= 0.000298, \text{ when } p = 0.01.$$

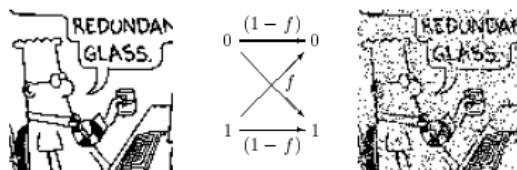
Much better than an uncoded system.

The price to pay: data rate 1/3.

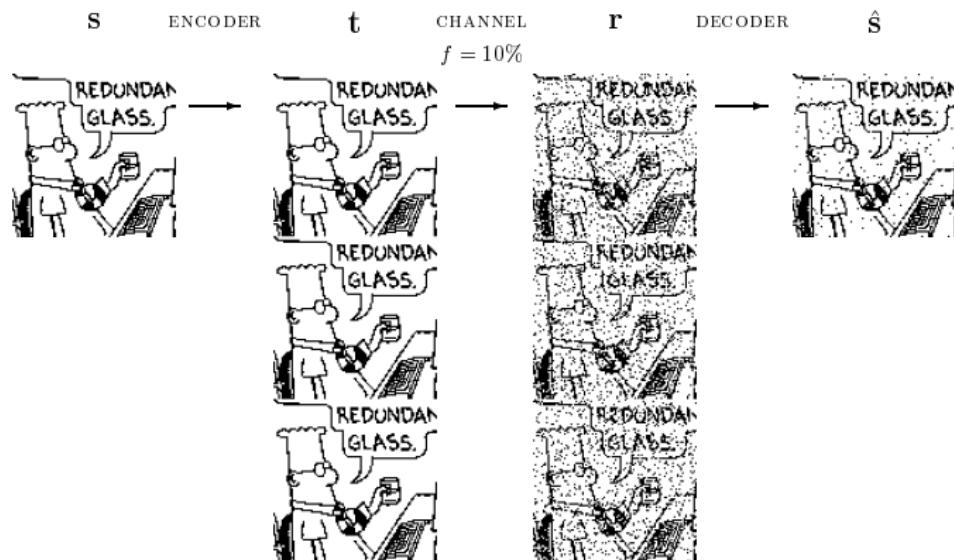
Coding theory: tradeoff between *error correction* and *data rate*.

Performance Comparison

Uncoded case ($f=0.1$)



Triple repetition code



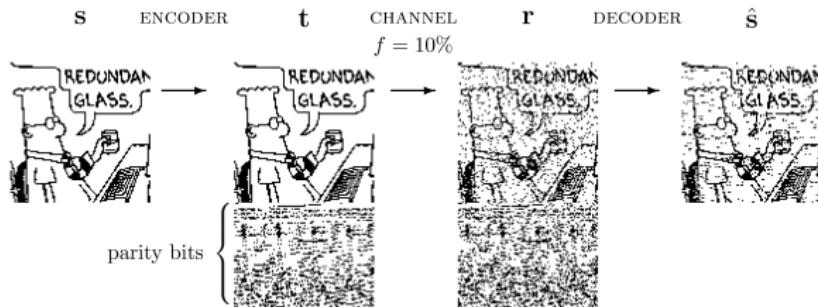
From David J.C. MacKay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.

The 2nd example: (7, 4) Hamming code

Encoding: encode every 4 bit information into 7 bits. (Details are omitted.)

Code rate: $r = 4/7 \approx 0.57$.

Much higher rate but slightly larger P_e .



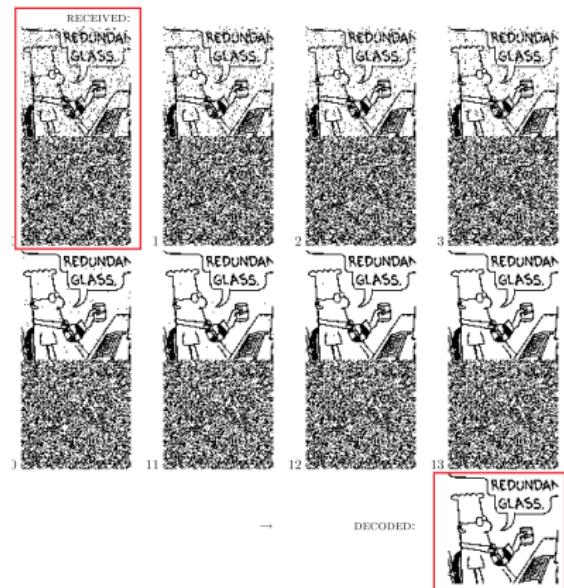
From David J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.

Another example - low-density parity-check code

Details are omitted here. Only simulation is presented

BSC with $p = 7.5\%$.

LDPC (20 000, 10 000) $r = 0.5$
iterative



From David J.C. MacKay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.

Distance: Definition

Definition 3.2 (Distance)

A distance d on a set \mathcal{X} is a function

$$d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$$

such that for all $x, y, z \in \mathcal{X}$, the following conditions hold:

► Positive definite:

$$d(x, y) \geq 0 \text{ where } "=" \text{ holds iff } x = y.$$

► Symmetry:

$$d(x, y) = d(y, x).$$

► Triangle inequality:

$$d(x, z) \leq d(x, y) + d(y, z).$$

In this course, d is also translation invariant, that is,

$$d(x, y) = d(x + z, y + z).$$

Examples of Commonly Used Distances

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be two vectors of length n , for example,

$$\mathbf{x} = [9, 1, 0], \quad \mathbf{y} = [6, 1, 4] \in \mathbb{R}^3$$

► ℓ_2 -norm distance: Euclidean distance d_2

$$\begin{aligned} d_2(\mathbf{x}, \mathbf{y}) &= \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \\ &= \sqrt{3^2 + 0^2 + 4^2} = 5. \end{aligned}$$

► ℓ_1 -norm distance: d_1

$$\begin{aligned} d_1(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^n |x_i - y_i| \\ &= 3 + 0 + 4 = 7. \end{aligned}$$

► Hamming distance: d_H

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^n \delta_{x_i \neq y_i} \\ &= 1 + 0 + 1 = 2, \end{aligned}$$

where $\delta_{x_i \neq y_i} = 1$ if $x_i \neq y_i$ and $\delta_{x_i \neq y_i} = 0$ if $x_i = y_i$.

Hamming Distance

Definition 3.3 (Hamming Distance)

*Hamming distance:
number of different elements.*

For $x, y \in \mathbb{F}^n$, the Hamming distance is given by

$$\begin{aligned} d_H(x, y) &= \sum_{i=1}^n \delta_{x_i \neq y_i} \\ &= |\{i : x_i \neq y_i\}| \end{aligned} .$$

Fact 3.4

Hamming distance is a well defined distance.

To prove this fact, the only non-trivial part is the triangle inequality.

Proof of the Triangle Inequality for Hamming Distance

1. **scalar case:** $d_H(x_i, z_i) \leq d_H(x_i, y_i) + d_H(y_i, z_i)$:

If $x_i = z_i$, then the equality holds obviously.

If $x_i \neq z_i$, LHS = 1. We have three cases:

$$\left. \begin{array}{l} y_i = x_i \Rightarrow y_i \neq z_i \\ y_i = z_i \Rightarrow y_i \neq x_i \\ y_i \neq x_i \text{ and } y_i \neq z_i \end{array} \right\} \Rightarrow \text{RHS} \geq 1.$$

2. **vector case:**

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{z}) &= \sum_{i=1}^n d_H(x_i, z_i) \\ &\leq \sum_{i=1}^n (d_H(x_i, y_i) + d_H(y_i, z_i)) \\ &= d_H(\mathbf{x}, \mathbf{y}) + d_H(\mathbf{y}, \mathbf{z}). \end{aligned}$$

Hamming Distance: Properties

Fact 3.5

Hamming distance is translation invariant:

$$d_H(\mathbf{x}_1, \mathbf{x}_2) = d_H(\mathbf{x}_1 + \mathbf{y}, \mathbf{x}_2 + \mathbf{y}).$$

Definition 3.6 (Weight)

A weight of a vector $\mathbf{x} \in \mathbb{F}_q^n$ is defined as its Hamming distance from the zero vector:

$$\underline{\text{wt}(\mathbf{x}) = d_H(\mathbf{x}, 0)}.$$

Example:

- ▶ $\mathbf{x} = [9, 1, 4], \mathbf{y} = [0, 1, 4] \Rightarrow d_H(\mathbf{x}, \mathbf{y}) = 1.$
- ▶ $\mathbf{x} = [1, 2, 1, 2, 1], \mathbf{y} = [2, 0, 2, 0, 2] \Rightarrow d_H(\mathbf{x}, \mathbf{y}) = 5.$
- ▶ $\mathbf{x} = [0, 1, 0, 1] \Rightarrow \text{wt}(\mathbf{x}) = 2.$

Decoding Techniques

Suppose that a codeword $\mathbf{c} \in \mathcal{C} \subset \mathbb{F}_q^n$ is transmitted and a word \mathbf{y} is received. The decoding function is defined as the mapping

$$\begin{aligned}\mathcal{D} : \mathbb{F}_q^n &\rightarrow \mathcal{C} \\ \mathbf{y} &\mapsto \hat{\mathbf{c}} \in \mathcal{C}.\end{aligned}$$

Popular decoding strategies include

Maximum likelihood decoding:

$$\hat{\mathbf{c}}_{ML} = \mathcal{D}_{ML}(\mathbf{y}) = \arg \max_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{y} \text{ received} | \mathbf{c} \text{ sent}).$$

Minimum distance decoding:

$$\hat{\mathbf{c}}_{MD} = \mathcal{D}_{MD}(\mathbf{y}) = \arg \min_{\mathbf{c} \in \mathcal{C}} d_H(\mathbf{y}, \mathbf{c}).$$

They are equivalent for many channels.

Equivalence Between ML and MD decoding

Theorem 3.7

Consider a memoryless binary symmetric channel (BSC) with cross-over probability $p < 1/2$. Then

$$\hat{c}_{ML} = \hat{c}_{MD}.$$

Proof: $\Pr(y|c) = \prod_{i=1}^n \Pr(y_i|c_i) = p^{d_H(y,c)} (1-p)^{n-d_H(y,c)}$
 $= \left(\frac{p}{1-p}\right)^{d_H(y,c)} \cdot (1-p)^n$

$$\because p < \frac{1}{2} \therefore \frac{p}{1-p} < 1 \quad \Pr(y \text{ received} | c \text{ sent}) = \prod_{i=1}^n \Pr(y_i \text{ received} | c_i \text{ sent})$$

$$\therefore \text{when } d_H(y,c) \text{ is minimised,} \quad = p^{d_H(y,c)} (1-p)^{n-d_H(y,c)}$$

$$\Pr(y|c) \text{ is maximised.} \quad = (1-p)^n \left(\frac{p}{1-p}\right)^{d_H(y,c)}.$$

That $p < 1/2$ implies that $p/(1-p) < 1$. Hence, $\Pr(y \text{ received} | c \text{ sent})$ is a monotonically decreasing function of $d_H(y, c)$. The maximum $\Pr(y|c)$ is achieved when $d_H(y, c)$ is minimized. ◇

Distance of a Code

Definition 3.8

The **distance** of a code \mathcal{C} is defined as

$$d_H(\mathcal{C}) = \min_{\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{C}, \mathbf{x}_1 \neq \mathbf{x}_2} d_H(\mathbf{x}_1, \mathbf{x}_2).$$

Notation: An (n, M, d) -code:

a code of codeword length n , size M , and distance d .

Example: Consider the binary code $n=5, M=3, d=2$.
 $\mathcal{C} = \{00000, 00111, 11111\}$.
 $r = \frac{n}{k} = \frac{5}{3}$

It is a binary $(5, 3, 2)$ -code.

Example: Consider the ternary code $n=6, M=3, d=3$.
 $\mathcal{C} = \{000000, 000111, 111222\}$.
 $r = \frac{n}{k} = 6$.

It is a ternary $(6, 3, 3)$ -code.

Error Detection

Error detector: if the received word $\mathbf{y} \in \mathcal{C}$, let $\hat{\mathbf{c}} = \mathbf{y}$ and claim no error; if $\mathbf{y} \notin \mathcal{C}$, claim that errors happened.

error: received word is not codeword (i.e. not in the dict).

Theorem 3.9

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be an (n, M, d) code. The above error detector detects every pattern of up to $d - 1$ many errors.

Proof:

1. Every pattern of $d - 1$ many errors are detectable. Since at most $d - 1$ many errors happened, $0 < d_H(\mathbf{c}, \mathbf{y}) < d := d(\mathcal{C})$ and $\mathbf{y} \notin \mathcal{C}$. The receiver will claim that errors happened.
2. Exists a pattern of d many errors that is not detectable. By the definition of the code distance, there exist $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}$ s.t. $d_H(\mathbf{c}_1, \mathbf{c}_2) = d$. Suppose that \mathbf{c}_1 is the transmitted codeword and the channel errors happen to be $\mathbf{e} = \mathbf{c}_2 - \mathbf{c}_1$ (d errors happened). Then $\mathbf{y} = \mathbf{c}_2$ is received. As $\mathbf{c}_2 \in \mathcal{C}$, the detector claims that no error happened and set $\hat{\mathbf{c}} = \mathbf{c}_2$.



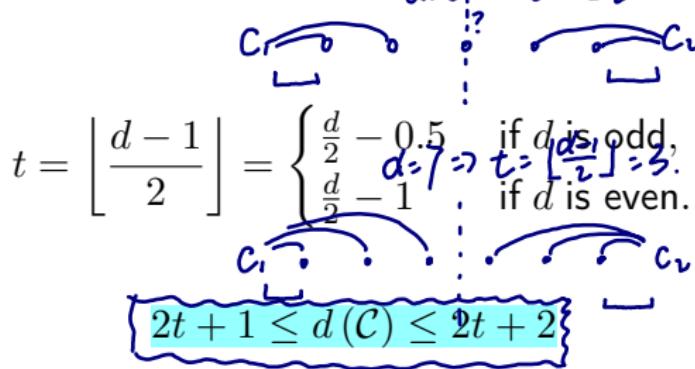
Error Correction

Theorem 3.10

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be an (n, M, d) code. The minimum distance decoder can correct every pattern of up to $t := \lfloor (d - 1) / 2 \rfloor$ many errors.

$$d=6 \Rightarrow t = \lfloor \frac{d-1}{2} \rfloor = 2$$

Side mark:



and

Examples:

The previous ternary $(6, 3, 3)$ code is exactly 1-error-detecting.

Error Correction: Proof

Proof: Let \mathcal{D} be the minimum distance decoder. Let \mathbf{c} and \mathbf{y} be the transmitted codeword and received word respectively. Let $\hat{\mathbf{c}} = \mathcal{D}_{MD}(\mathbf{y})$.

1. If $d_H(\mathbf{y}, \mathbf{c}) \leq t = \lfloor (d-1)/2 \rfloor$, then $\hat{\mathbf{c}} = \mathbf{c}$.

Suppose that $\hat{\mathbf{c}} \neq \mathbf{c}$. By the way the decoder \mathcal{D}_{MD} is defined,

$$d_H(\mathbf{y}, \hat{\mathbf{c}}) \leq d_H(\mathbf{y}, \mathbf{c}) \leq t.$$

On the other hand, by the definition of the code distance,

$$d \leq d_H(\mathbf{c}, \hat{\mathbf{c}}) \leq d_H(\mathbf{c}, \mathbf{y}) + d_H(\mathbf{y}, \hat{\mathbf{c}}) \leq 2t \leq d-1,$$

which is a contradiction.

$$\mathbf{y} \xrightarrow{t+1} \mathbf{d}$$

2. \exists a pair $(\mathbf{c}, \mathbf{y}) \in \mathcal{C} \times \mathbb{F}_q^n$ s.t. $d_H(\mathbf{y}, \mathbf{c}) = t+1$ and it may happen that $\mathcal{D}_{MD}(\mathbf{y}) \neq \mathbf{c}$.

By the definition of the code distance, $d_H(\mathbf{y}, \mathbf{c}) = t+1 \Rightarrow d_H(\mathbf{y}, \mathbf{c}') \geq d - (t+1)$

W.l.o.g., assume the first d positions of $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ are different. Define \mathbf{y}' such that $y'_i = c'_i$, $i = 1, 2, \dots, t+1$ and $y'_i = c_i$, $i = t+2, \dots, n$. It is clear that $d_H(\mathbf{y}', \mathbf{c}') = t+1 = d_H(\mathbf{y}, \mathbf{c}')$.

$d_H(\mathbf{y}, \mathbf{c}') = d - (t+1) \leq t+1 = d_H(\mathbf{y}, \mathbf{c})$. Hence, it may happen that $\hat{\mathbf{c}} = \mathcal{D}_{MD}(\mathbf{y}) \neq \mathbf{c}$. \diamond

Section 4

Linear Codes

- ▶ Definition.
 - ▶ Generator matrices.
 - ▶ Parity-check matrices.
- ▶ Decoding.

Remark: Why linear codes? Low complexity in encoding, decoding, and distance computation.

For the contents relevant to distance, Lin & Xing's book, Chapter 2, should be helpful.

Linear Codes: Definition

Block codes: all codewords are of the same length $\mathcal{C} \subset \mathbb{F}_q^n$.

Definition 4.1 (Linear Codes)

A linear code is a code for which any linear combination of codewords is also a codeword. (all zeros is always a codeword).

That is, let $\mathbf{u}, \mathbf{v} \in \mathcal{C} \subset \mathbb{F}_q^n$. Then $\lambda\mathbf{u} + \mu\mathbf{v} \in \mathcal{C}$, $\forall \lambda, \mu \in \mathbb{F}_q$.

Example of linear codes:

$$\mathcal{C} = \{\underbrace{0000, 0011, 1100, 1111}\}_{\text{in } \mathbb{F}_2^4}.$$

$$\mathcal{C} = \{\mathbf{v} \in \mathbb{F}_2^4 : \text{wt}(\mathbf{v}) \text{ is even.}\}.$$

Example of nonlinear codes:

$$\mathcal{C} = \{0000, 1100, 1111\}.$$

$$\begin{array}{r} \cancel{1} \cancel{1} \cancel{1} \\ \cancel{1} \cancel{0} \cancel{0} \\ \hline \cancel{2} \cancel{2} \cancel{1} \end{array} \times$$

$$\mathcal{C} = \{\mathbf{v} \in \mathbb{F}_3^4 : \text{wt}(\mathbf{v}) \text{ is even.}\}.$$

Definition 4.2 (Basis)

Let $\mathcal{B} = \{v_1, \dots, v_k\} \subset \mathbb{F}^n$. It is a basis of a set $\mathcal{C} \subset \mathbb{F}^n$ if it satisfies the following conditions:

► Linear independence property:

For all $\lambda_1, \dots, \lambda_k \in \mathbb{F}$, if $\sum \lambda_i v_i = \mathbf{0}$, then necessarily $\lambda_1 = \dots = \lambda_k = 0$.

► The spanning property:

For every $c \in \mathcal{C}$, there exist $\lambda_1, \dots, \lambda_k \in \mathbb{F}$ s.t. $c = \sum_i \lambda_i v_i$.

$\dim(\mathcal{C}) = k$: the # of vectors in a basis.

The basis \mathcal{B} is not unique in general, but the dimension is.

Example: Let $\mathcal{C} = \{0000, 0011, 1100, 1111\}$.

$\mathcal{B}_1 = \{0011, 1100\}$ is a basis for \mathcal{C} .

$\mathcal{B}_2 = \{0011, 1111\}$ is also a basis for \mathcal{C} .

$\dim(\mathcal{C}) = 2$.

Construct a Basis

Definition 4.3 (Linear Span)

For any subset $\mathcal{V} \subset \mathbb{F}^n$, define $\langle \mathcal{V} \rangle$ as the linear span of \mathcal{V} :

$$\langle \mathcal{V} \rangle = \left\{ \sum \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}, \mathbf{v}_i \in \mathcal{V} \right\}.$$

Construct a basis for a linear code $\mathcal{C} \subset \mathbb{F}^n$:

1. From \mathcal{C} , take a nonzero vector, say \mathbf{v}_1 .
2. Take a nonzero vector, say \mathbf{v}_2 , from $\mathcal{C} - \langle \{\mathbf{v}_1\} \rangle$.
3. Take a nonzero vector, say \mathbf{v}_3 , from $\mathcal{C} - \langle \{\mathbf{v}_1, \mathbf{v}_2\} \rangle$.
4. Continue this process, until $\mathcal{C} - \langle \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\} \rangle = \phi$.
5. Set $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$.

The Size of a Linear Code

Theorem 4.4

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be a linear code and $\dim(\mathcal{C}) = k$, then $|\mathcal{C}| = q^k$.

Proof:

1. $\dim(\mathcal{C}) = k \Rightarrow \exists$ a basis $\mathcal{B} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ for \mathcal{C} .
2. $|\mathcal{C}| \leq q^k$:

Definition of the basis suggests $\mathcal{C} = \langle \mathcal{B} \rangle = \left\{ \sum_{i=1}^k \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}_q \right\}$.

There are q^k many possible linear combinations. Hence, $|\mathcal{C}| \leq q^k$ (repetition may exist).

3. $|\mathcal{C}| = q^k$:

It suffices to show that there is no repetition.

Let $\boldsymbol{\lambda}^{(1)} \neq \boldsymbol{\lambda}^{(2)}$. Let $\mathbf{x}^{(1)} = \sum_{i=1}^k \lambda_i^{(1)} \mathbf{v}_i$ and $\mathbf{x}^{(2)} = \sum_{i=1}^k \lambda_i^{(2)} \mathbf{v}_i$.

Then $\mathbf{x}^{(1)} - \mathbf{x}^{(2)} = \sum_{i=1}^k (\lambda_i^{(1)} - \lambda_i^{(2)}) \mathbf{v}_i \neq \mathbf{0}$ by linear independence of \mathbf{v}_i 's and the fact that $\boldsymbol{\lambda}^{(1)} \neq \boldsymbol{\lambda}^{(2)}$.

There is no repetition in the set $\left\{ \sum_{i=1}^k \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}_q \right\}$. ◊

Generator Matrix

Definition 4.5 (Generator Matrix)

A **generator matrix** G for a linear code $\mathcal{C} \subset \mathbb{F}^n$ is a matrix whose **rows** form a basis for \mathcal{C} .

For a given (n, k) linear code $\mathcal{C} \subset \mathbb{F}^n$, it can be defined using its generator matrix $G \in \mathbb{F}^{k \times n}$.

The **encoding** function that maps information symbols to a codeword is given by

$$E : \mathbb{F}^k \rightarrow \mathcal{C} \subset \mathbb{F}^n$$
$$s \mapsto c = sG \in \mathcal{C}.$$

Remark:

Encoding of a linear code is efficient: vector-matrix product.

Encoding of a nonlinear code is via a look-up table and hence computationally less efficient.

Examples

Example 1: the $(3, 1)$ repetition code: $\mathbf{G} = [1 \ 1 \ 1]$.

Example 2: the $(7, 4)$ Hamming code.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Example 3: the generator matrix is not unique.

$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ and $\mathbf{G}' = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ generate the same code
 $\mathcal{C} = \{0000, 1010, 0101, 1111\} \subset \mathbb{F}_2^4$.

Dual Code

Definition 4.6 (Dual Code)

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be a non-empty code. Its **dual code** \mathcal{C}^\perp is defined as

$$\mathcal{C}^\perp = \left\{ \mathbf{v} \in \mathbb{F}_q^n : \mathbf{v}\mathbf{c}^T = \sum_i v_i c_i = 0 \text{ for all } \mathbf{c} \in \mathcal{C} \right\}.$$

Lemma 4.7

For any non-empty code $\mathcal{C} \subset \mathbb{F}_q^n$ (linear or nonlinear), its dual code \mathcal{C}^\perp is always linear.

Proof: Take arbitrary $\mathbf{v}_1, \mathbf{v}_2 \in \mathcal{C}^\perp$. Then for all $\lambda_1, \lambda_2 \in \mathbb{F}_q$ and for all $\mathbf{c} \in \mathcal{C}$,

$$(\lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2) \mathbf{c}^T = \lambda_1 \mathbf{v}_1 \mathbf{c}^T + \lambda_2 \mathbf{v}_2 \mathbf{c}^T = 0,$$

which implies $\lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2 \in \mathcal{C}^\perp$. ◊

Parity Check Matrix

Definition 4.8 (Parity-Check Matrix)

A **parity-check matrix** \mathbf{H} for a linear code $\mathcal{C} \subset \mathbb{F}_q^n$ is a **generator matrix for the dual code \mathcal{C}^\perp** .

For a code $\mathcal{C} [n, k]$, it holds that

- $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ and $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$.
- $\boxed{\mathbf{H} \cdot \mathbf{G}^T = \mathbf{0}}$.

Define a linear code via its parity-check matrix:

$$\mathcal{C} = \left\{ \mathbf{c} \in \mathbb{F}_q^n : \underline{\mathbf{c}\mathbf{H}^T = \mathbf{0}} \right\}.$$

Examples

$$H \cdot G^T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- The (3, 1) repetition code:

$$G = \begin{bmatrix} k \times n \\ 1 & 1 & 1 \end{bmatrix} \text{ and } H = \begin{bmatrix} (n-k) \times n \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}.$$

- The (7, 4) Hamming code:

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \text{ and } H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

- A self-dual code is a code s.t. $\mathcal{C} = \mathcal{C}^\perp$,

Example: $\mathcal{C} = \{0000, 1010, 0101, 1111\}$, where

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} = H.$$

Self-dual codes do not exist for vector space \mathbb{R}^n or \mathbb{C}^n .

Relation Between \mathbf{G} and \mathbf{H}

Consider $\mathcal{C}[n, k] \subset \mathbb{F}_q^n$. Write \mathbf{G} and \mathbf{H} in systematic forms:

- Let $\mathbf{G} = [\mathbf{I}_k \ \mathbf{A}] \in \mathbb{F}_q^{k \times n}$, where $\mathbf{A} \in \mathbb{F}_q^{k \times (n-k)}$.
- Let $\mathbf{H} = [\mathbf{B} \ \mathbf{I}_{n-k}] \in \mathbb{F}_q^{(n-k) \times n}$, where $\mathbf{B} \in \mathbb{F}_q^{(n-k) \times k}$.

Lemma 4.9

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

It holds that $\mathbf{B} = -\mathbf{A}^T$ $\mathbf{B} = -\mathbf{A}^T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$

Proof:

$$\begin{aligned} \mathbf{H}\mathbf{G}^T &= [\mathbf{B} \ \mathbf{I}_{n-k}] \begin{bmatrix} \mathbf{I}_k \\ \mathbf{A}^T \end{bmatrix} = \mathbf{B} \cdot \mathbf{I}_k + \mathbf{I}_{n-k} \cdot \mathbf{A}^T \\ &= -\mathbf{A}^T + \mathbf{A}^T = \mathbf{0} \in \mathbb{F}_q^{(n-k) \times k}. \end{aligned}$$

Systematic form:

Why? Easy to compute \mathbf{H} from \mathbf{G} , and vice versa.

How? Gaussian-Jordan elimination.

Hamming Weight

of nonzero components.

Hamming Weight of \mathbf{x} : $\text{wt}(\mathbf{x}) = |\{i : x_i \neq 0\}| = d(\mathbf{x}, \mathbf{0})$.

Theorem 4.10

For a linear code \mathcal{C} , $d_H(\mathcal{C}) = \min_{\mathbf{x} \in \mathcal{C} \setminus \{\mathbf{0}\}} \text{wt}(\mathbf{x})$.

Proof: $d_H(\mathbf{c}_1, \mathbf{c}_2) = \text{wt}(\mathbf{c}_1 - \mathbf{c}_2) = \text{wt}(\mathbf{c}')$ for some $\mathbf{c}' \in \mathcal{C}$ (by the definition of linear codes).

Notation: $\mathcal{C}[n, k, d]$: n : codeword length; k : dimension; d : distance.

Distance and Parity Check Matrix

Theorem 4.11

Let \mathcal{C} be a linear code defined by the parity-check matrix \mathbf{H} . Then that $d(\mathcal{C}) = d$ is equivalent to that

1. Every $d - 1$ columns of \mathbf{H} are linearly independent.
2. There exist d linearly dependent columns.

Two Confusing Concepts

spark: minimum # linearly dependent

rank: maximum # linearly independent

Given a matrix H ,

- spark: minimum number of linearly dependent columns
- column rank: maximum number of linearly independent columns.

Theorem 4.11 suggests that $\text{spark}(\mathbf{H}) = d(\mathcal{C})$

Example 4.12

- $\mathbf{H} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$: $\text{spark}(\mathbf{H}) = 3$ and column rank $(\mathbf{H}) = 2$.

- $\mathbf{H} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$ is an $n \times (n + 1)$ matrix:

$\text{spark}(\mathbf{H}) = 2$ and column rank $(\mathbf{H}) = n$.

Application of Theorem 4.11: Binary Hamming Codes

Definition 4.13 (Binary Hamming Codes)

The parity-check matrix of the **binary Hamming code**

$\mathcal{H}[2^r - 1, 2^r - 1 - r, 3]$ consists of all the nonzero binary vectors (columns) of length r (Also denoted by \mathcal{H}_r .)

Example 4.14

dimension = spark

The $\mathcal{H}_2[3, 1, 3]$ is given by

minimum number of columns

$$H = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 2 & 3 \end{bmatrix}, \text{ that are independent.}$$

and the $\mathcal{H}_3[7, 4, 3]$ is given by

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{bmatrix}.$$

The Distance of Binary Hamming Codes

Corollary 4.15

The distance of a binary Hamming code is 3, i.e., $d(\mathcal{H}_r) = 3$.

Proof: We apply Theorem 4.11.

- ▶ That there is no zero column implies that the minimum number of linearly dependent columns is at least 2, i.e., $d(\mathcal{C}) = \text{spark}(\mathbf{H}) \geq 2$.
- ▶ In the binary field, that every two columns are distinct implies that every two columns are linearly independent. Hence, $d(\mathcal{C}) = \text{spark}(\mathbf{H}) \geq 3$.
- ▶ It is easy to see that there exist three columns that are linearly dependent (for example the first three columns). Therefore $d(\mathcal{C}) = 3$.
◊

Corollary 4.16

Binary Hamming codes correct up to one error.

Theorem 4.11: Proof $d(\mathcal{C}) = d \Leftrightarrow$ every $d-1$ columns of \mathbf{H} are L.I.
 \exists d columns of \mathbf{H} are L.D.

Proof: Let \mathbf{h}_i be the i^{th} column of \mathbf{H} . $\forall \mathbf{c} \in \mathcal{C}$, let i_1, \dots, i_K be the locations where $c_i \neq 0$. By the definition of parity-check matrix,

$$\mathbf{0} = \sum_{i=1}^n c_i \mathbf{h}_i = \sum_{k=1}^K c_{i_k} \mathbf{h}_{i_k}.$$

$d(\mathcal{C}) = d \Rightarrow$ **Claim 2:** $d(\mathcal{C}) = d$ implies that $\exists \mathbf{c} \in \mathcal{C}$ s.t. $\text{wt}(\mathbf{c}) = d$. That is, $\sum_{k=1}^d c_{i_k} \mathbf{h}_{i_k} = \mathbf{0}$, or, $\mathbf{h}_{i_1}, \dots, \mathbf{h}_{i_k}$ are linearly dependent.

$d(\mathcal{C}) = d \Rightarrow$ **Claim 1:** Suppose not. $\exists \mathbf{h}_{i_1}, \dots, \mathbf{h}_{i_{d-1}}$ are linearly independent, i.e., $\sum_{k=1}^{d-1} x_{i_k} \mathbf{h}_{i_k} = \mathbf{0}$. Let $\mathbf{x} = [0 \cdots x_{i_1} \cdots x_{i_k} \cdots x_{i_{d-1}} \cdots 0]$. Then $\text{wt}(\mathbf{x}) \leq d-1$ and $\mathbf{x} \in \mathcal{C}$. Hence $d(\mathcal{C}) \leq d-1$. A contradiction with $d(\mathcal{C}) = d$.

Claims 1&2 $\Rightarrow d(\mathcal{C}) = d$: That every $d-1$ columns are linearly independent implies no nonzero codeword of weight $d-1$. That there exists d columns that are linearly dependent means the existence of a codeword of weight d . Hence $d(\mathcal{C}) = \min_{\mathbf{x} \in \mathcal{C} \setminus \{\mathbf{0}\}} \text{wt}(\mathbf{x}) = d$. \diamond

Syndrome Vector

Let $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ be a parity-check matrix of a linear code $\mathcal{C}[n, k] \subset \mathbb{F}_q^n$. Suppose that the received word is given by $\mathbf{y} \in \mathbb{F}_q^n$.

Define the **syndrome vector**

$$\mathbf{s} := \mathbf{y}\mathbf{H}^T = \mathbf{e}\mathbf{H}^T$$

It depends only on the error vector not the transmitted codeword.

In particular, let $\mathbf{y} = \mathbf{x} + \mathbf{e}$ where $\mathbf{x} \in \mathcal{C}$ is the transmitted codeword and $\mathbf{e} \in \mathbb{F}_q^n$ is the error vector introduced by the channel. It holds that

$$\mathbf{s} = \mathbf{y}\mathbf{H}^T = (\mathbf{x} + \mathbf{e})\mathbf{H}^T = \mathbf{e}\mathbf{H}^T.$$

Syndrome Decoding

Syndrome decoding:

1. compute syndrome vector, $s = yH^T = eH^T$
2. find error vector with minimum weight (M)

MD decoding: Find $\hat{c} = \arg \min_{c \in C} d_H(c, y)$.
 $\hat{e} = \arg \min_e \text{wt}(e)$
 $e = y - \hat{e}$.

Syndrome decoding:

1. For the received word y , compute the syndrome vector: $s := yH^T$.
2. Find the error vector e with the minimum weight: (MD decoding)

$$\hat{e} = \arg \min_e \text{wt}(e) \text{ s.t. } s = eH^T. \quad (1)$$

3. Decode y as $\hat{c} = y - \hat{e}$.

Comments: In general, it is computationally challenging to solve (1). However, all practical codes have particular structures in the parity-check matrix so that the decoding problem can be solved efficiently.

Decoding of Binary Hamming Codes

Take \mathcal{H}_3 (Definition 4.13) as an example.

index = 1 \Rightarrow 1st bit error

$$e = [1000000]$$

Assume that $y = [0111111]$. Find the MD decoded codeword $\hat{c} \in \mathcal{C}$.

$$S^T = H \cdot y^T = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \therefore \hat{c} = [1111111]$$

Since $d(\mathcal{H}_3) = 3$, it corrects up to 1 error.

For any e s.t. $\text{wt}(e) = 1$, let $e_i \neq 0$ for some $i \in [n]$. Then

$$s = eH^T = e_i h_i^T = h_i^T.$$

$$\underbrace{s^T}_{\text{red}} = \underbrace{H y^T}_{\text{red}} = \underbrace{H e^T}_{\text{red}} = \begin{array}{|c|c|} \hline \text{red bar} & \text{white bar} \\ \hline \end{array} = h_i$$

h_i

1

In the example, $s = [001]$, $e = [1000000]$ and $\hat{c} = [1111111]$.

Section 5

Coding Bounds

- ▶ Sphere packing (Hamming) bound
- ▶ Sphere covering (Gilbert-Varshamov) bound
- ▶ Singleton bound and MDS codes

The lectures will only cover sphere packing, sphere covering, singleton bounds and relevant contents. Reference: Lin & Xing's book, Chapter 5.

Coding Bounds: Motivation

Consider the Hamming code \mathcal{H}_r :

$$r = 2: [3, 1, 3]$$

$$r = 3: [7, 4, 3]$$

$$r = 4: [15, 11, 3]$$

Questions:

- ▶ Can we do better?
- ▶ What is the best that we can do?

(arger d. more corrections)

It is **possible** to construct linear codes with parameters

- ▶ $[7, 4, 4]$ over \mathbb{F}_8 .
- ▶ $[15, 11, 5]$ over \mathbb{F}_{16} .

Hamming Bound

Theorem 5.1 (Hamming bound, sphere-packing bound)

For a code of length n and distance d , the number of codewords is upper bounded by

$$M \leq q^n / \left(\sum_{i=0}^t \binom{n}{i} (q-1)^i \right),$$

volume of balls

where $t := \lfloor \frac{d-1}{2} \rfloor$. $\mathbb{F}_q^n = \{v\}$, $b = (x; v)$ positions
 $r=0$: center

points: $r=1 \Rightarrow |\{v : d_H(x, v) = 1\}| = \binom{n}{1} (q-1)$ parameters

$$r=2 \Rightarrow |\{v : d_H(x, v) = 2\}| = \binom{n}{2} (q-1)^2$$

∴ # balls = # of codewords

$$\vdots$$
$$r=t \Rightarrow |\{v : d_H(x, v) = t\}| = \binom{n}{t} (q-1)^t$$

$M = \frac{\text{available space}}{\text{ball volume}}$

$$\therefore \text{Volume} = \sum \text{points} = \sum_{i=0}^t \binom{n}{i} (q-1)^i$$
$$= \frac{q^n}{\sum_{i=0}^t \binom{n}{i} (q-1)^i}$$

Examples

Definition 5.2 (Perfect Codes)

A perfect code is a code that attains the Hamming bound.

- ▶ Binary Hamming code $\mathcal{H}_r [2^r - 1, 2^r - 1 - r, 3]$ is a perfect code.
 $d = 3 \Rightarrow t = \left\lfloor \frac{d-1}{2} \right\rfloor = 1$.
Ball Volume: $\sum_{i=0}^t \binom{n}{i} (q-1)^i = 1 + (2^r - 1) = 2^r$.
Hamming bound: $q^n / \sum_{i=0}^t \binom{n}{i} (q-1)^i = 2^{2^r-1} / 2^r = 2^{2^r-r-1} = 2^k$.
- ▶ Perfect codes are **rare** (binary Hamming codes & Golay codes).

Hamming Bound: Proof (1)

Define a ball in \mathbb{F}_q^n centered at $\mathbf{x} \in \mathbb{F}_q^n$ with radius t by

$$B(\mathbf{x}, t) = \{\mathbf{y} \in \mathbb{F}_q^n : d(\mathbf{x}, \mathbf{y}) \leq t\}.$$

Step one: the balls $B(\mathbf{c}, t)$, $\mathbf{c} \in \mathcal{C}$, are disjoint.

For all $\mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$, it holds that $B(\mathbf{c}, t) \cap B(\mathbf{c}', t) = \emptyset$.

For a $\mathbf{y} \in B(\mathbf{c}, t)$, then $\mathbf{y} \notin B(\mathbf{c}', t)$ for all $\mathbf{c}' \neq \mathbf{c}$.

By triangle inequality: $d \leq d_H(\mathbf{c}, \mathbf{c}') \leq d_H(\mathbf{c}, \mathbf{y}) + d_H(\mathbf{y}, \mathbf{c}')$.

Then

$$\begin{aligned} d_H(\mathbf{y}, \mathbf{c}') &\geq d - d_H(\mathbf{c}, \mathbf{y}) \geq d - t = d - \left\lfloor \frac{d-1}{2} \right\rfloor \\ &> \left\lfloor \frac{d-1}{2} \right\rfloor = t, \end{aligned}$$

which implies $\mathbf{y} \notin B(\mathbf{c}', t)$.

Hamming Bound: Proof (2)

Step two: Consider the union of these balls.

Clearly $\bigcup_{\mathbf{c} \in \mathcal{C}} B(\mathbf{c}, t) \subset \mathbb{F}_q^n$. Hence,

$$\text{Vol} \left(\bigcup_{\mathbf{c} \in \mathcal{C}} B(\mathbf{c}, t) \right) = \sum_{\mathbf{c} \in \mathcal{C}} \text{Vol}(B(\mathbf{c}, t)) \leq \text{Vol}(\mathbb{F}_q^n) = q^n,$$

where the first equality holds because the balls do not overlap.

The volume of each ball is

$$\text{Vol}(B(\mathbf{c}, t)) = \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Therefore

$$M \text{Vol}(B(\mathbf{c}, t)) \leq q^n \quad \Rightarrow \quad M \leq q^n / \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$



Gilbert-Varshamov Bound

Theorem 5.3 (Gilbert-Varshamov bound, sphere covering bound)

For given code length n and distance d , there exists a code such that

$$q^n / \text{Vol}(d-1) \leq M,$$

where $\text{Vol}(d-1) := \sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i$.

Comment: Different from the sphere packing bound, which holds for all codes, the sphere covering bound claims the existence of a code. That means, some badly designed codes may not satisfy this bound.

Gilbert-Varshamov Bound: Proof

It's proved by construction.

Let $M_0 = \lceil q^n / \text{Vol}(d-1) \rceil > 1$.

It suffices to show that exists a code with M_0 codewords.

Take an arbitrary word $\mathbf{c}_1 \in \mathbb{F}_q^n$.

Since $M_0 > 1$, or $q^n > \text{Vol}(d-1)$, it holds $\mathbb{F}_q^n \setminus B(\mathbf{c}_1, d-1) \neq \emptyset$.

Take an arbitrary word $\mathbf{c}_2 \in \mathbb{F}_q^n \setminus B(\mathbf{c}_1, d-1)$.

It is clear that $d(\mathbf{c}_1, \mathbf{c}_2) \geq d(\mathbf{c}_2 \notin B(\mathbf{c}_1, d-1))$.

Continue this process inductively.

Suppose to obtain codewords $\mathbf{c}_1, \dots, \mathbf{c}_{M_0-1}$ in this way.

Since $\text{Vol} \left(\bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \right) \leq (M_0 - 1) \text{Vol}(d-1) < q^n$,

it holds that $\mathbb{F}_q^n \setminus \bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \neq \emptyset$.

Take an arbitrary $\mathbf{c}_{M_0} \in \mathbb{F}_q^n \setminus \bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \neq \emptyset$.

Let $\mathcal{C} = \{\mathbf{c}_1, \dots, \mathbf{c}_{M_0}\}$.

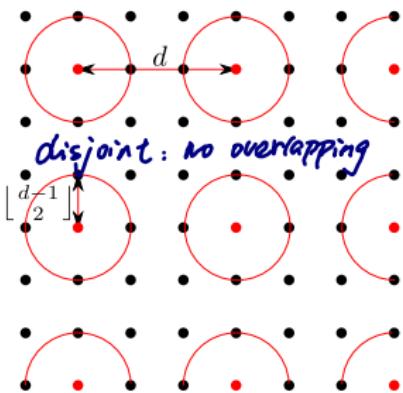
By construction, $d(\mathbf{c}, \mathbf{c}') > d-1$ for all $\mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$. Hence $d(\mathcal{C}) \geq d$. \diamond

Illustration for Sphere Packing and Covering

$$\frac{q_r^n}{\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i} \geq M \geq \frac{q_r^n}{\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i}$$

$$t = \lfloor \frac{d-1}{2} \rfloor$$

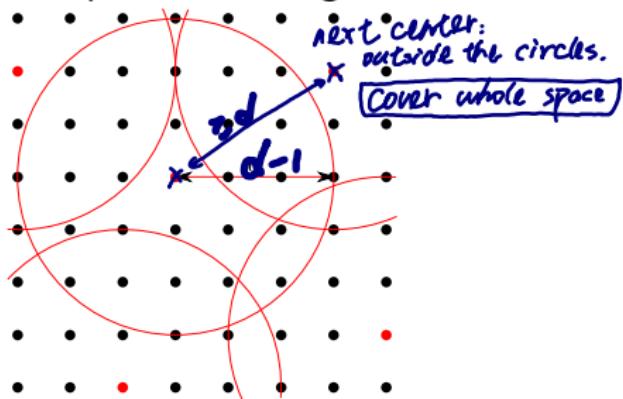
Sphere Packing



$$\text{Volume} = \sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i$$

$M \geq \frac{\text{available space}}{\text{ball volume}}$.

Sphere Covering



Singleton Bound and MDS

$$q^k \leq q^{n-d+1} \quad \begin{matrix} d-1 \text{ independent} \\ h-k=r \end{matrix}$$

Theorem 5.4 (Singleton Bound) $\therefore d = n - k + 1$ $H = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$ $n=7$ $r=3$ $k=4$

The distance of any code $\mathcal{C} \subset \mathbb{F}_q^n$ with M codewords satisfies

$$M \leq q^{n-d+1}.$$

In particular, if the code is linear and $M = q^k$, then

$$d \leq n - k + 1.$$

Definition 5.5 (MDS) MDS: $d = n - k + 1$

Codes that attain the singleton bound are maximum distance separable (MDS).

Binary Hamming codes $\mathcal{H}_r [2^r - 1, 2^r - 1 - r, 3]$ are not MDS in general.

- $d = 3 < n - k + 1 = r + 1$ for all $r \geq 3$.

Singleton Bound: Proof

Proof of the general case:

Let \mathcal{C} be of length n and distance d .

$\forall \mathbf{c} \in \mathcal{C}$, let $\mathbf{c}_{1:n-d+1} \in \mathbb{F}^{n-d+1}$ be the vector containing the first $n - d + 1$ entries of \mathbf{c} , and $\mathbf{c}_{n-d+2:n} \in \mathbb{F}^{d-1}$ be the vector composed of the last $d - 1$ elements of \mathbf{c} .

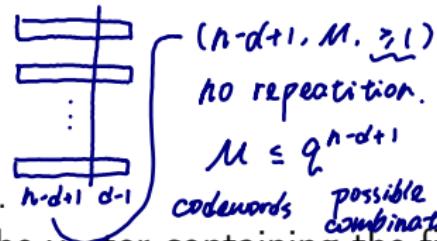
$\forall \mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$,

$$d \leq d_H(\mathbf{c}, \mathbf{c}') = d_H(\mathbf{c}_{1:n-d+1}, \mathbf{c}'_{1:n-d+1}) + d_H(\mathbf{c}_{n-d+2:n}, \mathbf{c}'_{n-d+2:n}).$$

But $d_H(\mathbf{c}_{n-d+2:n}, \mathbf{c}'_{n-d+2:n}) \leq d - 1$.

Hence, $d_H(\mathbf{c}_{1:n-d+1}, \mathbf{c}'_{1:n-d+1}) \geq d - (d - 1) = 1$.

The truncated codewords are all distinct. Hence, $M \leq q^{n-d+1}$. \diamond



Proof for linear codes:

Note that the parity-check matrix $\mathbf{H} \in \mathbb{F}^{(n-k) \times n}$ contains $n - k$ rows.

Every $n - k + 1$ columns must be linearly dependent.

By Theorem 4.11, $d \leq n - k + 1$. \diamond

Dual of MDS Codes

$$\text{MDS: } d = n - k + 1$$

$$C: (n, k, n-k+1)$$

$$C^\perp: \begin{array}{l} G_C \in \mathbb{F}^{k \times n} \\ H_C \in \mathbb{F}^{(n-k) \times n} \\ G_C^\perp \in \mathbb{F}^{\frac{(n-k)}{k+1} \times n} \end{array}$$

$$G_C^\perp = H_C \in \mathbb{F}^{(n-k) \times n}$$

$$d = n - k + 1$$

\Rightarrow every $d-1 = n-k$ columns of H are linearly independent.

Theorem 5.6

If a linear code C is MDS, then its dual code C^\perp is also MDS.

Let the linear code $C[n, k]$ be MDS.

According to Theorem 5.6, one has

$$H_C \in \mathbb{F}^{k \times n}$$

each codeword of C^\perp can be written as $S \in \mathbb{F}^{(1, n-k)}$

$$\begin{array}{c} S \\ \hline n-k \end{array} \quad \boxed{} \quad \boxed{}$$

	Parity-check matrix	Generator Matrix	Parameters
C	$H \in \mathbb{F}^{(n-k) \times n}$	$G \in \mathbb{F}^{k \times n}$	$(n, k, n - k + 1)$
C^\perp	$G \in \mathbb{F}^{k \times n}$	$H \in \mathbb{F}^{(n-k) \times n}$	$(n, n - k, k + 1)$

suppose $d < k+1 \rightarrow d=k$

Key for the proof: Theorem 4.11.

If $C[n, k]$ is MDS, then every set of $n - k$ columns of H is linear (so that $d = k$ possible).

$\therefore \exists c \in C^\perp$ s.t. at most k non-zeros

~~at least $n-k$ zeros~~

\therefore first part of H is full rank

$$c = sh$$

$\therefore s$ is all zero vector.

$\Rightarrow c$ is all zero.

Dual of MDS Codes (Theorem 5.6): Proof

Suppose $d(\mathcal{C}^\perp) < k + 1$. Then there exists a nonzero codeword $\mathbf{c} \in \mathcal{C}^\perp$ with at most k nonzero entries and at least $n - k$ zeros. Since permuting the coordinates reserves the codeword weights (i.e., the distance), w.l.o.g., assume that the last $n - k$ coordinates of \mathbf{c} are zeros.

Write the generator matrix of \mathcal{C}^\perp (the parity-check matrix of \mathcal{C}) as $\mathbf{H} = [\mathbf{A}, \mathbf{H}']$, where $\mathbf{A} \in \mathbb{F}^{(n-k) \times k}$ and $\mathbf{H}' \in \mathbb{F}^{(n-k) \times (n-k)}$. By definition of the generator matrix, there exists $\mathbf{s} \in \mathbb{F}^{n-k}$ such that $\mathbf{c} = \mathbf{s}\mathbf{H}$.

As \mathcal{C} is MDS, by Theorem 4.11 the columns of \mathbf{H}' are linearly independent. That is, \mathbf{H}' is invertible. That the last $n - k$ coordinates of \mathbf{c} are zeros implies that $\mathbf{s} = \mathbf{c}_{k+1:n} (\mathbf{H}')^{-1} = \mathbf{0}$. But $\mathbf{s} = \mathbf{0}$ implies $\mathbf{c} = \mathbf{s}\mathbf{H} = \mathbf{0}$ which contradicts the assumption that $\mathbf{c} \neq \mathbf{0}$. Hence, $d(\mathcal{C}^\perp) \geq k + 1$. By the Singleton bound, $d(\mathcal{C}^\perp) = k + 1$. \diamond

Section 6

RS & BCH Codes

- ▶ Reed-Solomon Codes
 - ▶ Definition and properties.
 - ▶ Decoding
- ▶ Cyclic and BCH codes

The contents in this section are significant re-organization and condensation of the materials of many sources, including Lin & Xing's book, Chapters 7 and 8, and Roth's book, Chapters 5, 6 and 8.

Reed-Solomon Codes



Our Heroes: [Irving S. Reed](#) and [Gustave Solomon](#)

Used in

- ▶ Magnetic recording (all computer hard disks use RS codes)
- ▶ Digital versatile disks (CDs, DVDs, etc.)
- ▶ Optical fiber networks (ITU-TG.795)
- ▶ ADSL transceivers (ITU-TG.992.1)
- ▶ Wireless telephony (3G systems, 4G systems)
- ▶ Digital satellite broadcast (ETS 300-421S, ETS 300-429)
- ▶ Deep space exploration (all NASA probes)

RS Codes: Evaluation Mapping

Definition 6.1 (Evaluation Mapping)

Let \mathbb{F}_q be a finite field. Let $n \leq q - 1$ (typically $n = q - 1$).

Let $\mathcal{A} = \{\alpha_1, \dots, \alpha_n\} \subset \mathbb{F}_q$.

For any polynomial $f(x) \in \mathbb{F}_q[x]$, define the evaluation mapping $\text{eval}(f(x))$ that maps f to a vector $\mathbf{c} \in (\mathbb{F}_q)^n$

$$\mathbb{F}_7 = \{0, 1, \dots, 6\}, \quad \mathcal{A} = \{1, \alpha, \dots, \alpha^5\} = \{1, 3, 2, 6, 4, 5\}$$

$\begin{cases} f \mapsto \mathbf{c} = [c_1, \dots, c_n] \text{ where } c_i = f(\alpha_i) \\ \text{e.g. } f(x) = 2x+1 \end{cases}$

Example 6.2 $C = \text{eval}(f) = \{2x+1, 2x^3+1, 2x^2+1, 6, 1, 4\}$

$$\begin{matrix} C_i &= f(\alpha^i) \\ &= 3 &= 0 &= 5 \end{matrix}$$

$\mathbb{F}_7 = \{0, 1, 2, 3, 4, 5, 6\}$. Choose the primitive element $\alpha = 3$.

Let $\mathcal{A} = \{1, \alpha, \dots, \alpha^5\} = \{1, 3, 2, 6, 4, 5\}$.

$f(x) = 2x + 1$, $\mathbf{c} = \text{eval}(f) = [3, 0, 5, 6, 2, 4]$.

$f(x) = 3x^2 + x + 2$, $\mathbf{c} = \text{eval}(f) = [6, 4, 2, 4, 5, 5]$.

② $f_1(x) = 3x^2 + x + 2$

$$C = \text{eval}(f_1) = \{3x^2+1+2, 3x^3+3+2, 2, 4, 5, 5\}$$
$$\begin{matrix} &= 6 &= 4 \end{matrix}$$

RS Codes: Definition $\mathbb{F}_q^{[k]}$ degree $k-1$
 $f(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$, $a_i \in \mathbb{F}_q$.
 $|f(x)| = q^k$

$$\mathbb{F}_q = \{0, 1, \dots, p-1, \dots\} = \{0, 1, \alpha, \dots, \alpha^{q-2}\}$$

$$\mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\} = \{1, \alpha, \dots, \alpha^{q-2}\}$$

Definition 6.3 (Reed-Solomon Codes) $\cup c_i = f(\alpha^i)$

Given $A = \{\alpha_1, \dots, \alpha_n\} \subset \mathbb{F}_q$, an $[n, k]$ q -ary RS code
 $C = \{[f(1), f(\alpha), \dots, f(\alpha^{q-2})] : \deg(f) \leq k-1\}$

$C = \{\text{eval}(f), 0 \leq \deg f \leq k-1\}$.

The set A is called a defining set of points of C .
 $|C| = q^{k-1} \cdot k \cdot ? \rightarrow n-k+1$
 length dimension $= q-k$ distance (MDS)

A common choice of defining set of points of C is $A = \{1, \alpha, \dots, \alpha^{q-2}\}$
 where α is a primitive element in \mathbb{F}_q .

In this case, $n = q - 1$.

RS Codes: Properties

Theorem 6.4

1. RS codes are linear codes.
2. RS codes are MDS, i.e., The distance of the RS code is $d = n - k + 1$.

Proof:

1. Let $c_1 = \text{eval}(f_1)$ and $c_2 = \text{eval}(f_2)$ where $\deg f_1 \leq k - 1$ and $\deg f_2 \leq k - 1$. Then $\alpha c_1 + \beta c_2 = \text{eval}(g)$ with $g = \alpha f_1 + \beta f_2$. Since $\deg g \leq k - 1$, $\text{eval}(g) \in \mathcal{C}$.
2. A polynomial of degree $\leq k - 1$ can have at most $k - 1$ zeros. Hence, $\forall c \in \mathcal{C} \text{ s.t. } c \neq 0$, $c = \text{eval}(f)$ has weight at least $n - k + 1$.
polynomial of degree $k-1 \Rightarrow$ at most $k-1$ zeros for each codeword
⇒ non-zero element at least $n-k+1$
⇒ distance = $n-k+1$

RS Codes: Conventional Definition

$$C = S \cdot G$$

Theorem 6.5

$$[f(1), f(\alpha), \dots, f(\alpha^{n-1})] = [a_0, a_1, \dots, a_{k-1}]$$

Let the defining set of points is $\{1, \alpha, \dots, \alpha^{n-1}\}$ with order $(\alpha) = n$ (typically $n = q - 1$). The generated RS code has generator matrix and parity-check matrix given by

$$G = \left[\begin{array}{cccc} 1 & 1 & \cdots & 1 \\ 1 & \alpha & \cdots & \alpha^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \cdots & \alpha^{(k-1)(n-1)} \end{array} \right]$$

α - primitive element of \mathbb{F}_q .

and

$$H = \left[\begin{array}{cccc} 1 & \alpha & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & \cdots & \alpha^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{n-k} & \cdots & \alpha^{(n-k)(n-1)} \end{array} \right]$$

Generator Matrix: Justification

For any $\mathbf{c} \in \mathcal{C}$, there exists a polynomial $f(x) = a_0 + a_1x + \cdots + a_{k-1}x^{k-1}$ s.t. $\mathbf{c} = \text{eval}(f) = [f(1), f(\alpha), \dots, f(\alpha^{n-1})] \in \mathcal{C}$.

Note that $\forall i \in [n]$, $c_i = f(\alpha^i) = \sum_{\ell=0}^{k-1} a_\ell (\alpha^{i-1})^\ell = [a_0, \dots, a_{k-1}] \mathbf{G}_i$ where \mathbf{G}_i is the i -th column of the \mathbf{G} matrix.

One has $\mathcal{C} = \{\mathbf{s}\mathbf{G} : \mathbf{s} \in \mathbb{F}_q^k\}$ and \mathbf{G} is a generator matrix of \mathcal{C} .

Remark: In the definition of the generator matrix (Def. 4.5), the rows of \mathbf{G} are required to be linearly independent. We shall prove it later.

Parity-Check Matrix: Justification

$$A_{ij} = \sum_{\ell=1}^k (\ell^{\text{th}} \text{ row of } G) \cdot (j^{\text{th}} \text{ row of } M)$$

$= \sum_{\ell=1}^k \alpha^{(\ell-1)(i-1)} \cdot \alpha^{(i-1)j}$

Lemma 6.6

$$G = \begin{bmatrix} 1 & \alpha & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & \cdots & \alpha^{(n-k)(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^k & \cdots & \alpha^{(n-k)(n-1)} \end{bmatrix}^T$$

$$H = \begin{bmatrix} 1 & \alpha & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & \cdots & \alpha^{(n-k)(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^k & \cdots & \alpha^{(n-k)(n-1)} \end{bmatrix}^T$$

The G and H matrices defined in Theorem 6.5 satisfy $GH^T = 0$.

$$= \sum_{\ell=1}^k \alpha^{(\ell+i-j-1)} \alpha^{(i-1)j} = \frac{\alpha^{(i+j-1)n} - 1}{\alpha^{i+j-1} - 1} = 0$$

Proof: Let $A := GH^T \in \mathbb{F}_q^{k \times (n-k)}$.

$\forall i \in [k]$ and $\forall j \in [n-k]$, it holds that

$$\begin{aligned} A_{i,j} &= \sum_{\ell=1}^n \alpha^{(\ell-1)(i-1)} \alpha^{(\ell-1)j} = \sum_{\ell=1}^n \alpha^{(i+j-1)(\ell-1)} \\ &\stackrel{(a)}{=} \frac{\alpha^{(i+j-1)n} - 1}{\alpha^{i+j-1} - 1} \stackrel{(b)}{=} 0, \end{aligned}$$

where (a) comes from that $i + j - 1 < n$ and $\alpha^{i+j-1} \neq 1$, and (b) holds because $\alpha^n = 1$.

Row Rank of the \mathbf{G}/\mathbf{H} Matrix

Theorem 6.7

The rows of the \mathbf{G}/\mathbf{H} matrix in Theorem 6.5 are linearly independent.

Proof: It is sufficient to show that any k -column sub-matrix of \mathbf{G} ($(n - k)$ -column sub-matrix of \mathbf{H}) has full rank.

Note that a k -column sub-matrix of \mathbf{G} is of the form

$$\mathbf{G}' = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha^{i_1} & \alpha^{i_2} & \cdots & \alpha^{i_k} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(k-1)i_1} & \alpha^{(k-1)i_2} & \cdots & \alpha^{(k-1)i_k} \end{bmatrix},$$

which is a Vandermonde matrix (defined and analysed later). A Vandermonde matrix has full rank. Hence the rows of \mathbf{G} are linearly independent. \diamond

Vandermonde Matrix

Definition 6.8 (Vandermonde Matrix)

A Vandermonde matrix $\mathbf{V} \in \mathbb{F}^{n \times n}$ is of the form

$$\mathbf{V} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} \end{bmatrix}.$$

Theorem 6.9

The determinant of a Vandermonde matrix $\mathbf{V} \in \mathbb{F}^{n \times n}$ is

$$|\mathbf{V}| = \prod_{i < j} (\alpha_j - \alpha_i).$$

As a result, if $\alpha_i \neq \alpha_j$, $1 \leq i \neq j \leq n$, then $|\mathbf{V}| \neq 0$ and \mathbf{V} is of full rank.

Determinant: A Recap

Definition 6.10 (Determinant)

$\forall \mathbf{A} \in \mathbb{F}^{n \times n}$, its determinant $|\mathbf{A}|$ is computed via

$$|\mathbf{A}| = \sum_{j=1}^n (-1)^{i+j} a_{i,j} |\mathbf{M}_{i,j}|,$$

where $\mathbf{M}_{i,j}$ is the **minor matrix** obtained by deleting row i and column j from \mathbf{A} .

Lemma 6.11

1. $|\mathbf{AB}| = |\mathbf{A}| |\mathbf{B}|$.
2. If \mathbf{B} results from \mathbf{A} by adding a multiple of one row/column to another row/column, then $|\mathbf{B}| = |\mathbf{A}|$.
3. $|\mathbf{A}| \neq 0 \Leftrightarrow \mathbf{A}$ is of full rank.

Theorem 6.9: Proof (1)

We prove Theorem 6.9 by using induction.

Recall that

$$\mathbf{V}_n = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_{n-1} & \alpha_n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha_1^{n-2} & \alpha_2^{n-2} & \cdots & \alpha_{n-1}^{n-2} & \alpha_n^{n-2} \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_{n-1}^{n-1} & \alpha_n^{n-1} \end{bmatrix}$$

Let $(\mathbf{V}'_n)_{:,2} = (\mathbf{V}_n)_{:,2} - (\mathbf{V}_n)_{:,1}$, \dots , $(\mathbf{V}'_n)_{:,n} = (\mathbf{V}_n)_{:,n} - (\mathbf{V}_n)_{:,1}$. We obtain
minus first column

$$\mathbf{V}'_n = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ \alpha_1 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha_1^{n-2} & \alpha_2^{n-2} - \alpha_1^{n-2} & \cdots & \alpha_{n-1}^{n-2} - \alpha_1^{n-2} & \alpha_n^{n-2} - \alpha_1^{n-2} \\ \alpha_1^{n-1} & \alpha_2^{n-1} - \alpha_1^{n-1} & \cdots & \alpha_{n-1}^{n-1} - \alpha_1^{n-1} & \alpha_n^{n-1} - \alpha_1^{n-1} \end{bmatrix}$$

$$\begin{aligned} & (\alpha_2^{n-1} - \alpha_1^{n-1}) - \alpha_1(\alpha_2^{n-2} - \alpha_1^{n-2}) \\ &= \alpha_2^{n-1} - \alpha_1 \alpha_2^{n-2} = \alpha_2^{n-2} (\alpha_2 - \alpha_1) \end{aligned}$$

Theorem 6.9: Proof (2)

Let $(\mathbf{V}_n'')_{n,:} = (\mathbf{V}_n')_{n,:} - \alpha_1 (\mathbf{V}_n')_{n-1,:}, \dots, (\mathbf{V}_n'')_{2,:} = (\mathbf{V}_n')_{2,:} - \alpha_1 (\mathbf{V}_n')_{1,:}$. We obtain

$$\mathbf{V}_n'' = \left[\begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ 0 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \alpha_2^{n-2} - \alpha_1 \alpha_2^{n-3} & \cdots & \alpha_{n-1}^{n-2} - \alpha_1 \alpha_{n-1}^{n-3} & \alpha_n^{n-2} - \alpha_1 \alpha_n^{n-3} \\ 0 & \alpha_2^{n-1} - \alpha_1 \alpha_2^{n-2} & \cdots & \alpha_{n-1}^{n-1} - \alpha_1 \alpha_{n-1}^{n-2} & \alpha_n^{n-1} - \alpha_1 \alpha_n^{n-2} \end{array} \right]$$

$$= \left[\begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ 0 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & (\alpha_2 - \alpha_1) \alpha_2^{n-3} & \cdots & (\alpha_{n-1} - \alpha_1) \alpha_{n-1}^{n-3} & (\alpha_n - \alpha_1) \alpha_n^{n-3} \\ 0 & (\alpha_2 - \alpha_1) \alpha_2^{n-2} & \cdots & (\alpha_{n-1} - \alpha_1) \alpha_{n-1}^{n-2} & (\alpha_n - \alpha_1) \alpha_n^{n-2} \end{array} \right]$$

$$= \left[\begin{array}{c} 1 \\ \mathbf{V}_{n-1}(\alpha_2, \dots, \alpha_n) \end{array} \right] \left[\begin{array}{ccccc} 1 & & & & \\ & \alpha_2 - \alpha_1 & & & \\ & & \ddots & & \\ & & & \alpha_{n-1} - \alpha_1 & \\ & & & & \alpha_n - \alpha_1 \end{array} \right]$$

Hence $|\mathbf{V}_n| = |\mathbf{V}'_n| = |\mathbf{V}''_n| = |\mathbf{V}_{n-1}| \prod_{j>1} (\alpha_j - \alpha_1)$.



Decoding with Known Error Locations

Let e be the error vector.

Let $\mathcal{I} = \{i : e_i \neq 0\}$ be the set of **error locations**.

$e_{\mathcal{I}}$, $H_{\mathcal{I}}$: sub-vector and sub-matrix of e and H respectively.

$$H_I y_I^T = H_I e_I^T = s^T$$

If we knew error locations \mathcal{I} :

Solve $H_{\mathcal{I}} e_{\mathcal{I}}^T = s^T$. ($e_{\mathcal{I}}^T = H_{\mathcal{I}}^\dagger s^T$)

full rank

Complexity of pseudo-inverse $H_{\mathcal{I}}^\dagger$: $O(d^3)$.

Erasure Correction

Recall the erasure channel model.

Suppose that $\mathbf{c} \in \mathcal{C}$ was transmitted.

Receive $\mathbf{r} = [c_1 \cdots c_{i-1} ? c_{i+1} \cdots c_n]$ (at most $d - 1$ symbols erased).

Decoding: Set the missing symbols to zero, i.e., $\mathbf{r}_{\mathcal{I}} = \mathbf{0}$.

Then $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where $\mathbf{e}_{\mathcal{I}^c} = \mathbf{0}$.

$$\mathbf{s}^T = \mathbf{H}\mathbf{r}^T = \mathbf{H}\mathbf{c}^T + \mathbf{H}\mathbf{e}^T = \mathbf{H}_{\mathcal{I}}\mathbf{e}_{\mathcal{I}}^T.$$

$$\begin{bmatrix} \alpha^{i_1-1} & \alpha^{i_2-1} & \dots & \alpha^{i_s-1} \\ \alpha^{2(i_1-1)} & \alpha^{2(i_2-1)} & \dots & \alpha^{2(i_s-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{s(i_1-1)} & \alpha^{s(i_2-1)} & \dots & \alpha^{s(i_s-1)} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_s \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_s \end{bmatrix}.$$

A Specific Example

Consider a $[7, 4, 4]$ RS code over \mathbb{F}_8 ($\mathbb{F}_2[x]/x^3 + x + 1$).

Let α be a primitive element (a root of $f(x) = x^3 + x + 1$).
 $\alpha \cdot 1^3 + \alpha \cdot 1^2 + 1$

$$\mathbb{F}_8: \alpha^3 = \alpha + 1$$

$$\begin{array}{ccccccccc} 000 & 001 & 010 & 100 & 011 & 110 & 111 & 101 \\ 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \end{array}$$

Encoded message $m(x) = \alpha x^3 + \alpha x^2 + x$.

$$c = \text{eval}(m) = [1 \ \alpha^5 \ \alpha \ 1 \ \alpha^5 \ \alpha^6 \ \alpha^5].$$

$$r = [1 \ \alpha^5 \ \alpha \ 1 \ ? \ ? \ \alpha^5].$$

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \end{bmatrix}$$

$$cH^T = 0 \Rightarrow \begin{bmatrix} \alpha^4 & \alpha^5 \\ \alpha & \alpha^3 \\ \alpha^5 & \alpha \end{bmatrix} \begin{bmatrix} c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} \alpha \\ 1 \\ \alpha \end{bmatrix} \text{ matrix inverse} \begin{bmatrix} c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} \alpha^5 \\ \alpha^6 \end{bmatrix}.$$

Error Correction

In previous example:

- ▶ “Error” (erasure) locations are known.
- ▶ The error values are found via matrix inverse.

For error correction:

- ▶ Find error locations
 - ▶ Exhaustive search: complexity $\binom{n}{t} = O(n^t)$.
 - ▶ In practice, \exists methods to find error locations efficiently.
- ▶ Correct errors with given error locations
 - ▶ Methods to avoid matrix inverse.

Efficient Error Correction: Definitions

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$

$$\text{size}(H) = n-k \times n$$

Definitions 6.12 $d = n-k+1$

Syndrome polynomial: $\tilde{h} = q_n - 1$

$$S(z) = \sum_{j=0}^{n-k-1} s_j z^j, \text{ where } s = rH^T = eH^T.$$

Received
 $XH^T = 0$
Construct $S(z)$ with r .

Error locator polynomial: (information about error locations)

$L(z) = 0 \text{ if } z = \alpha^{-i}$: error locations.
 $L(z) = \prod_{i \in \mathcal{I}} (1 - \alpha^i z).$ try all α^k ($q_n - 1$ times)
"n"

Error evaluator polynomial: (information about errors)

$E(z)$ is only 1 term if $z = \alpha^{-i}$ error symbol at error locations.
 $E(z) = L(z) \sum_{i \in \mathcal{I}} \frac{e_i \alpha^i}{(1 - \alpha^i z)} = \sum_{i \in \mathcal{I}} e_i \alpha^i \prod_{j \in \mathcal{I} \setminus \{i\}} (1 - \alpha^j z).$

$$E(z) = \prod_{i \in \mathcal{I}} (1 - \alpha^i z) \sum_{i \in \mathcal{I}} \frac{e_i \alpha^i}{(1 - \alpha^i z)} = \sum_{i \in \mathcal{I}} e_i \alpha^i \prod_{j \in \mathcal{I} \setminus \{i\}} (1 - \alpha^j z)$$

Remark: The receiver can compute the syndrome vector and the syndrome polynomial easily.

Information Encoded in $L(z)$ and $E(z)$

- ▶ If we know $L(z)$, we can find error locations.
 - ▶ $\begin{cases} L(\alpha^{-k}) = 0 & \text{if } k \in \mathcal{I} \\ L(\alpha^{-k}) \neq 0 & \text{if } k \notin \mathcal{I} \end{cases}$.
 - ▶ Error locations can be found by exhaustively computing $L(\alpha^{-k})$, $0 \leq k \leq \underline{n-2}$.
- ▶ $E(z)$ helps find errors e_i , $i \in \mathcal{I}$, without matrix inverse.
 - ▶ $\forall k \in \mathcal{I}$, $E(\alpha^{-k}) = e_k \alpha^k \prod_{j \neq k} (1 - \alpha^j \alpha^{-k}) \neq 0$.
 - ▶ $e_k = E(\alpha^{-k}) / \left(\alpha^k \prod_{j \neq k} (1 - \alpha^j \alpha^{-k}) \right)$.
 - ▶ or $e_k = -E(\alpha^{-k}) / \frac{d}{dz} L(\alpha^{-k})$, where $\frac{d}{dz} L(z)$ is the derivative of $L(z)$.
 - ▶ Complexity is reduced from $O(n^3)$ to $O(n^2)$.

Decoding strategy: from $S(z)$ to find $L(z)$ and $E(z)$.

An Example of $L(z)$ and $E(z)$

- ▶ $\mathcal{I} = \{1, 2, 5\}$.
- ▶ $L(z) = (1 - \alpha z)(1 - \alpha^2 z)(1 - \alpha^5 z)$
 - ▶ $L(z) = 0$ if $z = \alpha^{-1}, \alpha^{-2}$, or α^{-5} .
 - ▶ $L(z) \neq 0$ otherwise.
- ▶ $E(z) = e_1 \alpha^1 (1 - \alpha^2 z)(1 - \alpha^5 z)$ T1
 $+ e_2 \alpha^2 (1 - \alpha z)(1 - \alpha^5 z)$ T2
 $+ e_5 \alpha^5 (1 - \alpha z)(1 - \alpha^2 z)$ T3

	T1	T2	T3	$E(z)$
$z = \alpha^{-1}$	$\neq 0$	$= 0$	$= 0$	$\neq 0$
$z = \alpha^{-2}$	$= 0$	$\neq 0$	$= 0$	$\neq 0$
$z = \alpha^{-5}$	$= 0$	$= 0$	$\neq 0$	$\neq 0$

Properties of $L(z)$ and $E(z)$

Let $t = \lfloor \frac{d-1}{2} \rfloor$. $\alpha^{-k} \mid_{k \in \mathbb{Z}} \Rightarrow L(z) = 0$ $\Rightarrow L(z), E(z)$ do not share common roots.
 $E(z)$ is 1 term. $\neq 0$

Theorem 6.13

1. $\gcd(L(z), E(z)) = 1$.
2. *The key equation:*

$$\boxed{E(z) = L(z) S(z) \bmod z^{d-1}}$$

3. (*Uniqueness*) Let $a(z), b(z) \in \mathbb{F}_q[z]$ be such that $\deg(a(z)) \leq t - 1$, $\deg(b(z)) \leq t$, $\gcd(a(z), b(z)) = 1$ and

$$a(z) \equiv S(z) b(z) \pmod{z^{d-1}}.$$

Then $a(z)$ and $b(z)$ are unique up to a constant.

That is, we can treat $a(z) = cE(z)$, $b(z) = cL(z)$, and $E(z)$ and $L(z)$ are generated from an error vector e s.t. $\text{wt}(e) \leq t$.

$$\text{Decoding Process } E(z) = L(z)S(z) \bmod z^{d-1} \quad \because E(z) \text{ is linear combination of } S(z), z^{d-1}$$

$$= L(z)S(z) + f(z)z^{d-1}$$

$$\therefore E(z) = \gcd(S(z), z^{d-1})$$

1. Compute the syndrome vector and polynomial s and $S(z)$ respectively.

2. Apply Euclidean algorithm to z^{d-1} and $S(z)$, i.e.,

$$z^{d-1} = q_1(z)S(z) + r_1(z)$$

$$S(z) = q_2(z)r_1(z) + r_2(z)$$

⋮

$$r_{\ell-2}(z) = q_\ell(z)r_{\ell-1}(z) + r_\ell(z),$$

where $\deg(r_\ell(z)) \leq t - 1$.

3. By Bézout's Identity (Lem. 1.5), one has

$$r_\ell(z) = a(z)S(z) + b(z)z^{d-1} \equiv a(z)S(z) \bmod z^{d-1}.$$

4. Let c be the leading coefficient of the polynomial $a(z)$, i.e., $c^{-1}a(z)$

is a monic polynomial. By Theorem 6.13, set

highest order $L(z) = c^{-1}a(z)$, and $E(z) = c^{-1}r_\ell(z)$.

5. Find the error locations $i \in \mathcal{I}$ from $L(z)$ and the errors e_i from $E(z)$.

$$\hat{\mathbf{c}} = \mathbf{y} - \mathbf{e}.$$

The complexity is highly reduced!

Theorem 6.13, Part 1: Proof

Proof: $L(z)$ has roots α^{-i} , $i \in \mathcal{I}$. None of them is a root of $E(z)$.
 $L(z)$ and $E(z)$ does not share any roots.
 $\gcd(L(z), E(z)) = 1$.

Theorem 6.13, Part 2: Proof

Theorem 6.13 part 2 is a direct consequence of the lemma below.

$$\begin{aligned} S(z) &= \sum_{j=0}^{d-2} s_j z^j = \sum_{j=0}^{d-2} \sum_{i \in \mathcal{I}} e_i \alpha^{i(j+1)} z^j \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{d-2} (\alpha^i z)^j \right) \quad \downarrow \alpha^d z^d \bmod z^{d-1} = 0 \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{\infty} (\alpha^i z)^j \right) \bmod z^{d-1} \end{aligned}$$

Lemma 6.14

$$S(z) \equiv \sum_{i \in \mathcal{I}} \frac{e_i \alpha^i}{1 - \alpha^i z} \bmod z^{d-1}$$

Proof: As $s = rH^T = eH^T$, it follows that $\frac{\sum_{i \in \mathcal{I}} e_i \alpha^i}{1 - \alpha^i z}$.

Hence, $s_j = \sum_{i=0}^{n-1} e_i \alpha^{i(j+1)} = \sum_{i \in \mathcal{I}} e_i \alpha^{i(j+1)}$, $\forall 0 \leq j \leq d-2$.

By the definition of $S(z)$, it holds that

$$\begin{aligned} S(z) &= \sum_{j=0}^{d-2} s_j z^j = \sum_{j=0}^{d-2} \sum_{i \in \mathcal{I}} e_i \alpha^{i(j+1)} z^j \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{d-2} (\alpha^i z)^j \right) \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{\infty} (\alpha^i z)^j \right) \bmod z^{d-1} \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \frac{1}{1 - \alpha^i z}. \end{aligned}$$



Theorem 6.13, Part 3: Proof

Proof: To prove the uniqueness, we assume that there exist

$(E(z), L(z)) \neq (E'(z), L'(z))$ s.t.

$$E(z) = S(z)L(z) \bmod z^{d-1} \text{ and } E'(z) = S(z)L'(z) \bmod z^{d-1}.$$

It follows that

$$\begin{aligned} E(z)L'(z) &= S(z)L(z)L'(z) \bmod z^{d-1} \\ &= E'(z)L(z) \bmod z^{d-1}. \end{aligned} \tag{2}$$

By assumption, $\deg(E(z)) \leq t - 1$ and $\deg(L'(z)) \leq t$.

It is clear that $\deg(E(z)L'(z)) \leq 2t - 1 \leq d - 2$.

The same is true for $E'(z)L(z)$.

As a result, (2) becomes

$$E(z)L'(z) = E'(z)L(z) \quad \left| \begin{array}{l} E(z) | E'(z)L(z) \\ E'(z) | E(z)L'(z) \\ \gcd(E(z), L(z)) = 1 \\ \gcd(E'(z), L'(z)) = 1 \end{array} \right. \Rightarrow \left| \begin{array}{l} E'(z) | E(z) \\ E(z) | E'(z) \end{array} \right.$$

Note $\gcd(E(z), L(z)) = 1$. By Lemma 1.12, $E(z) | E'(z)$ and

$L(z) | L'(z)$.

Similarly from $\gcd(E'(z), L'(z)) = 1$, $E'(z) | E(z)$ and $L'(z) | L(z)$.

Hence, $E(z) = cE'(z)$ and $L(z) = cL'(z)$ for some nonzero $c \in \mathbb{F}_q$. \diamond

An Example

Example: Consider the $[7, 3]$ RS code over \mathbb{F}_8 (\mathbb{F}_8 is given as follows).

0	1	α	α^2	α^3	α^4	α^5	α^6
000	001	010	100	011	110	111	101

Let the received signal be $y = [\alpha^3, \alpha, 1, \alpha^2, 0, \alpha^3, 1]$. Find \hat{c} .

Solutions to the Example

1. **Parameters:** $n - k = 4$, $d = 5$ ($t = 2$), and $\mathbf{H} \in \mathbb{F}_8^{4 \times 7}$.

$$\mathbf{H} = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \\ 1 & \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 \end{bmatrix}.$$

2. **Syndromes:** $\mathbf{s} = \mathbf{y}\mathbf{H}^T = [\alpha^3, \alpha^4, \alpha^4, 0]$.

$$S(z) = \alpha^4 z^2 + \alpha^4 z + \alpha^3.$$

3. **Key polynomials:** apply Euclidean algorithm to z^4 and $S(z)$.

3.1 $z^4 = (\alpha^3 z^2 + \alpha^3 z + \alpha^5) S(z) + (z + \alpha)$.

3.2 $L'(z) = \alpha^3 z^2 + \alpha^3 z + \alpha^5$. $E'(z) = z + \alpha$.

3.3 $L(z) = \alpha^5 z^2 + \alpha^5 z + 1$. $E(z) = \alpha^2 z + \alpha^3$.

4. **Find $\hat{\mathbf{c}}$:**

4.1 Plug $1, \alpha^{-1}, \dots$ into $L(z)$. $L(\alpha^{-2}) = L(\alpha^{-3}) = 0$.

4.2 According $E(z)$, we have $e_2 = \alpha^3$ and $e_3 = \alpha^6$.

4.3 $\hat{\mathbf{c}} = \mathbf{y} - \mathbf{e} = \mathbf{y} + \mathbf{e} = [\alpha^3, \alpha, \alpha, 1, 0, \alpha^3, 1]$.

Towards Cyclic and BCH Codes

Have seen

- ▶ Binary Hamming codes: $d = 3$.
- ▶ Reed-Solomon codes: MDS ($d = n - k + 1$) and requires large fields (typically $q = n + 1$).

Will introduce cyclic codes

- ▶ Reed-Solomon codes are a special case of cyclic codes.
- ▶ BCH codes as another special case.
 - ▶ Systematic way to construct binary codes with large distance.

Cyclic Codes

Definition 6.15

An $[n, k]$ linear code is **cyclic**

if for every codeword $\mathbf{c} = c_0c_1 \cdots c_{n-2}c_{n-1}$,

the right cyclic shift of \mathbf{c} , $c_{n-1}c_0c_1 \cdots c_{n-2}$, is also a codeword.

Example: The $\mathcal{H}[7, 3]$ has the parity-check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Its dual code \mathcal{H}_3^\perp (view \mathbf{H} as the generator matrix) is cyclic.

(The codewords are 1011100, 0101110, 0010111, 1110010, 1001011, 0111001, 1100101, 0000000.)

Generating Function

Definition 6.16

The generating function of a codeword $c = [c_0 \dots c_{n-1}]$ is

$$c(x) = c_0 + c_1x + \dots + c_{n-1}x^{n-1}. \quad \text{codewords} \leftrightarrow \text{polynomials} \quad (1-1)$$

It will be convenient to use $c(x)$ to represent a codeword c .

The right cyclic shift $c = (c_0, \dots, c_{n-1}) \mapsto c' = (c_{n-1}, c_0, \dots, c_{n-2})$ can be obtained by $c'(x) = x \cdot c(x) \bmod x^n - 1$ as

$$\begin{aligned} x \cdot c(x) &= c_0x + c_1x^2 + \dots + c_{n-2}x^{n-1} + c_{n-1}x^n \\ &= \cancel{c_{n-1}} + \underbrace{c_0x + c_1x^2 + \dots + c_{n-2}x^{n-1}}_{\text{still codeword}} \bmod x^n - 1. \end{aligned}$$

$$\frac{x^n - 1}{C_{n-1}x^n}$$
$$\frac{C_n \cdots x_n - C_{n-1}}{C_{n-1}}$$

Lemma 6.17

Let $c(x) \in \mathcal{C}$. For an arbitrary $u(x)$, $u(x)c(x) \bmod x^n - 1$ is in \mathcal{C} .

Generator Polynomial

cyclic code

- $c(x) \in \mathcal{C}$, arbitrary $u(x) \Rightarrow u(x)c(x) \bmod x^{n-1} \in \mathcal{C}$.
- exist $g(x) \Rightarrow u(x)g(x) = c(x)$ for all $c(x)$

Theorem 6.18

For a cyclic code \mathcal{C} , \exists a **unique** monic polynomial $g(x)$ s.t.
for all $c(x) \in \mathcal{C}$, $c(x) = u(x)g(x)$ for some $u(x)$.

Proof:

Let $g(x) \in \mathcal{C}$ be the nonzero polynomial of least degree.

Since \mathcal{C} is linear, w.l.o.g., assume that $g(x)$ is monic.

Then $\forall c(x) \in \mathcal{C}$, write $c(x) = u(x)g(x) + r(x)$.

By definition of cyclic codes, $u(x)g(x) \in \mathcal{C}$.

Hence, $r(x) \in \mathcal{C}$ by linearity of \mathcal{C} .

But $\deg(r(x)) < \deg(g(x))$, which implies $r(x) = 0$.

The uniqueness of $g(x)$ can be proved by contradiction. Suppose that there are two *monic* polynomials $g_1(x) \neq g_2(x)$ of the same degree that both generate \mathcal{C} . Then $g_1(x) - g_2(x) \in \mathcal{C}$ and $\deg(g_1 - g_2) < \deg(g_1)$, which forces $g_1(x) - g_2(x) = 0$. \diamond

Properties of the Generator Polynomial

Corollary 6.19

$$g(x) | x^n - 1.$$

$$\begin{aligned} q(x)g(x) + r(x) \bmod x^n - 1 &\in \mathcal{C}(x) \\ \therefore q(x)g(x) \bmod x^n - 1 &\in \mathcal{C}(x) \end{aligned}$$

Proof: Write $x^n - 1 = q(x)g(x) + r(x)$.

Take "mod $x^n - 1$ " on both sides.

$$0 = x^n - 1 \bmod x^n - 1 \in \mathcal{C}. \quad q(x)g(x) \bmod x^n - 1 \in \mathcal{C}(x).$$

$$\text{Hence } r(x) \bmod x^n - 1 \in \mathcal{C} \Rightarrow r(x) \in \mathcal{C} \Rightarrow r(x) = 0.$$

$$\deg(r(x)) < \deg(g(x))$$

Remark: Let $n = q^m - 1$.

We know how to factor $x^n - 1$ in terms of minimal polynomials.

$g(x)$ must be a product of minimal polynomials.

Generator Matrices of Cyclic Codes

Theorem 6.20

The **generator matrix** of a cyclic code $\mathcal{C} [n, k]$:

$$G = \begin{bmatrix} 1 & g(x) & & & \\ \text{row } xg(x) & & & & \\ \vdots & & & & \\ x^{k-1}g(x) & & & & \end{bmatrix} = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} \\ g_0 & g_1 & \cdots & g_{n-k} \\ \ddots & & & \ddots \\ g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}.$$

Component with max degree = $x^{n-1} = x^{k-1}g(x)$

Observations: $\therefore \deg(g(x)) = n - k$.

- ▶ $\deg(g(x)) = n - k$.
- ▶ Easy for implementation: can be implemented by using flip-flops.

Parity-Check Matrices of Cyclic Codes

Recall $g(x) | x^n - 1$. Define $h(x)$ such that $\underbrace{g(x) h(x)}_{\deg(1-k)} = \underbrace{x^n - 1}_{\deg(k)}$. Then $h(x)$ is a *monic polynomial with degree k*.

Write $h(x) = \sum_{i=0}^k a_i x^i$.

Definition 6.21

The **reciprocal polynomial** $h_R(x)$ of $h(x)$ is given by

$$h_R(x) = a_k + a_{k-1}x + \cdots + a_0x^k = x^k h(1/x).$$

Example: $h(x) = 1 + x^2 + x^3 \Rightarrow h_R(x) = 1 + x + x^3$.

$$\begin{array}{r} 1011 \\ \xrightarrow{\hspace{1cm}} \quad \Rightarrow \quad \xleftarrow{\hspace{1cm}} 1101 \end{array}$$

Parity-Check Matrix

Theorem 6.22

The parity-check matrix of the cyclic code $\mathcal{C} [n, k]$ is

$$H = \underbrace{\begin{bmatrix} 1 & h_R(x) & & & & h_0 \\ 0 & xh_R(x) & & & & h_0 \\ \vdots & & & & & \ddots \\ 0 & & & & & h_0 \\ x^{n-k-1}h_R(x) & & & & & h_0 \end{bmatrix}}_{\text{n rows}} = \begin{bmatrix} h_k & h_{k-1} & \cdots & h_0 \\ h_k & h_{k-1} & \cdots & h_0 \\ \ddots & & & \ddots \\ h_k & h_{k-1} & \cdots & h_0 \end{bmatrix}.$$

Corollary 6.23

The dual of a cyclic code, \mathcal{C}^\perp , is also cyclic.

$h_0^{-1}h_R(x)$ is the generator polynomial of \mathcal{C}^\perp .

Theorem 6.22: Proof

By assumption, $x^n - 1 = g(x)h(x)$. Note that

$$\begin{aligned} g(x)h(x) &= \left(\sum_{i=0}^{n-k} g_i x^i\right) \left(\sum_{i=0}^k h_i x^i\right) \\ &= \sum_{i=0}^n \left(\sum_{\ell=0}^i g_\ell h_{i-\ell}\right) x^i = \sum_{i=0}^n a_i x^i, \end{aligned}$$

coeff. can be written as convolution.

where $a_0 = g_0 h_0 = -1$, $a_n = g_{n-k} h_k = 1 \cdot 1 = 1$, and

$$a_i = \sum_{\ell=0}^i g_\ell h_{i-\ell} = 0, \quad 1 \leq i \leq n-1.$$

Let $\mathbf{A} = \mathbf{G}\mathbf{H}^T$ with

$$\mathbf{A}_{i,j} = [\underbrace{0, \dots, 0}_{i-1}, g_0, \dots, g_{n-k}, 0, \dots, 0] \cdot [\underbrace{0, \dots, 0}_{j-1}, h_k, \dots, h_0, 0, \dots, 0]^T.$$

It can be verified that $A_{1,1} = a_k$, $A_{1,2} = a_{k+1}$, \dots , and

$$\mathbf{A} = \mathbf{G}\mathbf{H}^T = \begin{bmatrix} a_k & a_{k+1} & \cdots & a_{n-1} \\ a_{k-1} & a_k & \cdots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_{n-k} \end{bmatrix} = \mathbf{0} \in \mathbb{F}^{k \times (n-k)} \quad \diamond$$

Cyclic Codes: An Example

To construct a cyclic code on \mathbb{F}_q , we realize that

- ▶ $M^{(i)}(x) \in \mathbb{F}_q[x]$
- ▶ $M^{(i)}(x) | x^{q^m-1} - 1.$

Definition 6.24

A **BCH code** over \mathbb{F}_q of length $n = q^m - 1$ is the cyclic code generated by
$$g(x) = \text{lcm}(M^{(a)}(x), \dots, M^{(a+\delta-2)}(x))$$

for some integer a . (The code is called **narrow-sense** if $a = 1$.)

Lemma 6.25

A **BCH code** defined in Definition 6.24 has $d \geq \delta$.

δ is referred to the designed distance.

Distance of BCH Codes: Proof of Lemma 6.25

Let α be the primitive element in \mathbb{F}_{q^m} . By construction, $\alpha^a, \dots, \alpha^{a+\delta-2}$ are roots of the generator polynomial $g(x)$.

That is, $\forall c \in \mathcal{C}$, the generating function $c(x)$ satisfies $c(\alpha^i) = 0$, $a \leq i \leq a + \delta - 2$. In matrix format,

$$\begin{bmatrix} 1 & \alpha^a & \dots & \alpha^{a(n-1)} \\ 1 & \alpha^{a+1} & \dots & \alpha^{(a+1)(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{a+\delta-2} & \dots & \alpha^{(a+\delta-1)(n-1)} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \mathbf{0} \quad (3)$$

Any $(\delta - 1)$ -column submatrix is a Vandermonde matrix and hence of full rank. This implies $d \geq \delta$. \diamond

Remark: The matrix in (3) is in $\mathbb{F}_{q^m}^{(\delta-1) \times n}$ while the vector $c \in \mathcal{C} \subset \mathbb{F}_q^n$. Hence the matrix is not a parity-check matrix when $m > 1$.

Example: Reed-Solomon Codes

Recall that a RS-code $\mathcal{C} [n, k, n - k + 1]$ is built on \mathbb{F}_q with typically $n = q - 1$.

Compare the parity-check matrix of a RS-code (Theorem 6.5) and Equation (3). It is clear that a RS-code is a special case of a BCH code with $m = 1$.

In particular, suppose that we are asked to build a BCH code over \mathbb{F}_q with $n = q - 1$ and $d \geq \delta = n - k + 1$. We find $M^{(i)}(x) \subset \mathbb{F}_q[x]$, $1 \leq i \leq 1 + \delta - 2 = n - k$. Since $M^{(i)}(x) \subset \mathbb{F}_q[x]$, it follows that $M^{(i)}(x) = x - \alpha^i$. Hence $g(x) = \prod_{i=1}^{n-k} (x - \alpha^i)$ and the generator matrix can be constructed (in a different form of that in Theorem 6.5) and good for implementation). Its parity-check matrix is given by the matrix in Equation (3). RS decoder can be directly applied for decoding.

Example: Binary BCH Codes

We have learned binary Hamming codes. The distance is always 3. The question is how to construct a binary code with large distance.

For example, how to construct a binary code of length 15 and $d \geq 5$?

1. For binary codes, use \mathbb{F}_2 . $n = 15 = 2^4 - 1$ hence $m = 4$.
2. $\delta = 5$ implies $g(x) = \text{lcm}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x))$.
3. The relevant cyclotomic cosets of 2 modulo 15 include

$\mathcal{C}_1 = \{1, 2, 4, 8\}$ and $\mathcal{C}_3 = \{3, 6, 9, 12\}$. Hence

$M^{(1)}(x) = \prod_{i \in \mathcal{C}_1} (x - \alpha^i) = M^{(2)}(x) = M^{(4)}(x)$ and
 $M^{(3)}(x) = \prod_{i \in \mathcal{C}_3} (x - \alpha^i)$. Furthermore,

$$g(x) = M^{(1)}(x) \cdot M^{(3)}(x).$$

4. Find the generator matrix and parity-check matrix according to Theorems 6.20 and 6.22 respectively.

From Hamming to BCH

Example: A binary code of length 15 and $d \geq 5$?

$$\begin{aligned}g(x) &= \text{lcm}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)) \\&= \text{lcm}(M^{(1)}(x), M^{(3)}(x)) \\&= M^{(1)}(x) \times M^{(3)}(x).\end{aligned}$$

RS codes are special cases of BCH codes ($m = 1$).

Have learned [7, 4, 3] Hamming code.

$$\boldsymbol{H} = \left[\begin{array}{ccccccc} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{array} \right]$$

Another view:

Let α be a primitive element of \mathbb{F}_8 that satisfies $\alpha^3 = \alpha + 1$.

The parity check matrix can be written as

$$\left[\begin{array}{ccccccc} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \end{array} \right].$$

(Column vectors in $\mathbb{F}_2^3 \mapsto$ Elements in \mathbb{F}_8)

From Hamming to BCH: Larger Distance

Binary BCH codes with $d \geq 5$:

$$\begin{aligned}g(x) &= \text{lcm} (M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)) \\&= \text{lcm} (M^{(1)}(x), M^{(3)}(x)).\end{aligned}$$

It holds that

$$\left[\begin{array}{ccccccc} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \\ 1 & \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 \end{array} \right] \mathbf{c} = \mathbf{0}$$

$$\text{But } c(\alpha) = 0 \Rightarrow \begin{cases} c(\alpha^2) = c(\alpha)^2 = 0 \\ c(\alpha^4) = c(\alpha)^4 = 0 \end{cases}$$

$$\mathbf{H} = \left[\begin{array}{ccccccc} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \end{array} \right]$$

Eventually, we get a $[7, 1, 7]$ code $\mathcal{C} = \{0000000, 1111111\}$.

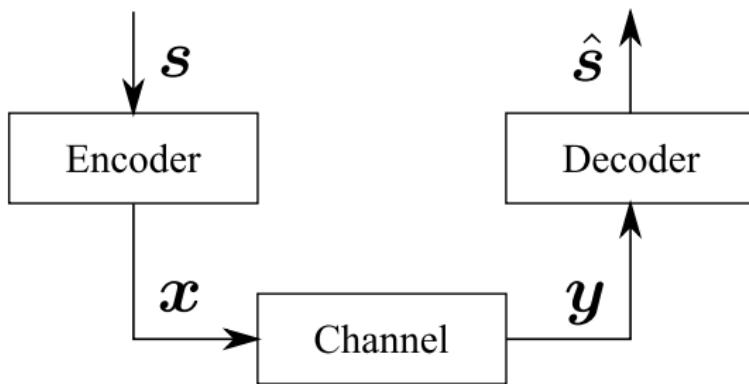
Section 3

Error Correcting Codes (ECC): Fundamentals

- ▶ Communication systems and channel models
- ▶ Definition and examples of ECCs
- ▶ Distance

For the contents relevant to distance, Lin & Xing's book, Chapter 2, should be helpful.

Communication Systems

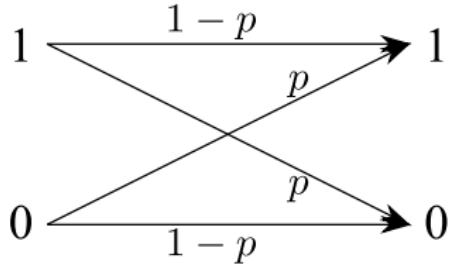


Abstract Channel Model: Binary Symmetric Channel (BSC)

Binary Symmetric Channel: a memoryless channel such that

$$\Pr(0 \text{ received} | 1 \text{ sent}) = \Pr(1 \text{ received} | 0 \text{ sent}) = p,$$

$$\Pr(1 \text{ received} | 1 \text{ sent}) = \Pr(0 \text{ received} | 0 \text{ sent}) = 1 - p.$$



p is called the transition (cross-over) probability.

Memoryless channel: A channel that satisfies

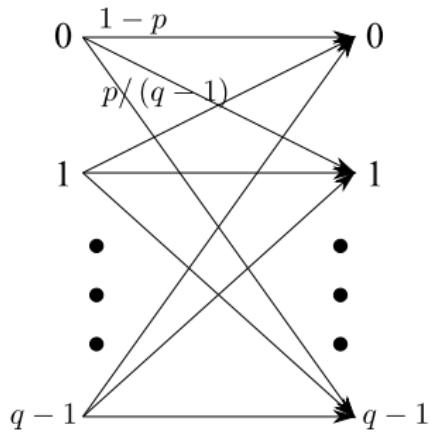
$$\Pr(y \text{ received} | x \text{ sent}) = \prod_{i=1}^n \Pr(y_i \text{ received} | x_i \text{ sent}).$$

The Memoryless q -ary Symmetric Channel

Define an **alphabet** set \mathbb{F}_q .

Both channel input x_i and channel output y_i are from \mathbb{F}_q .

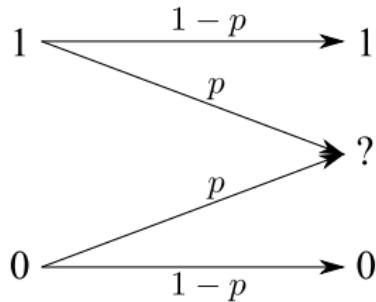
Crossover probability p :



$$\Pr(y_i|x_i) = \begin{cases} 1 - p & \text{if } y_i = x_i \\ p / (q - 1) & \text{if } y_i \neq x_i \end{cases}$$

The Memoryless Binary Erasure Channel (BEC)

Binary Erasure Channel:



- ▶ Internet traffic: a package got lost.
- ▶ Cloud storage: one copy of file messed up.

What is a Code?

Definition 3.1 (Code)

A code is a set \mathcal{C} containing (row) vectors of elements from \mathbb{F}_q .

An (n, M) block code: $\mathcal{C} \subset \mathbb{F}_q^n$ and $|\mathcal{C}| = M$.

A codeword: a vector in \mathcal{C} .

Codeword length: n

Code size: M

Dimension: $k = \log_q M$.

Rate: $r = k/n$.

Example 1:

$$\mathbb{F}_2 = \{0, 1\}. \mathcal{C} = \{0000, 1100, 1111\}.$$

$$n = 4. M = 3. k = \log_2 3 = 1.585. r = 0.3962.$$

Example 2:

$$\mathbb{F}_3 = \{0, 1, 2\}. \mathcal{C} = \{00000, 12121, 20202\}.$$

$$n = 5. M = 3. k = \log_3 3 = 1. r = 0.2.$$

Triple Repetition Code

Encoding

$$1 \rightarrow 111$$

$$0 \rightarrow 000$$

Decoding: majority voting

$$111, 110, 101, 011 \rightarrow 1$$

$$000, 001, 010, 100 \rightarrow 0$$

Error probability computation:

$$P(\hat{s} = 1 | s = 0)$$

$$= P(111|0) + P(110|0) + P(101|0) + P(011|0)$$

$$= p^3 + 3p^2(1-p)$$

$$= 0.000298, \text{ when } p = 0.01.$$

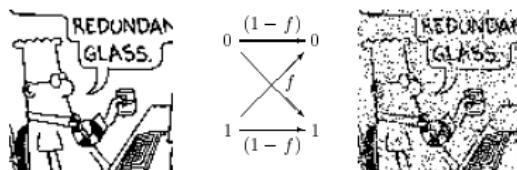
Much better than an uncoded system.

The price to pay: data rate 1/3.

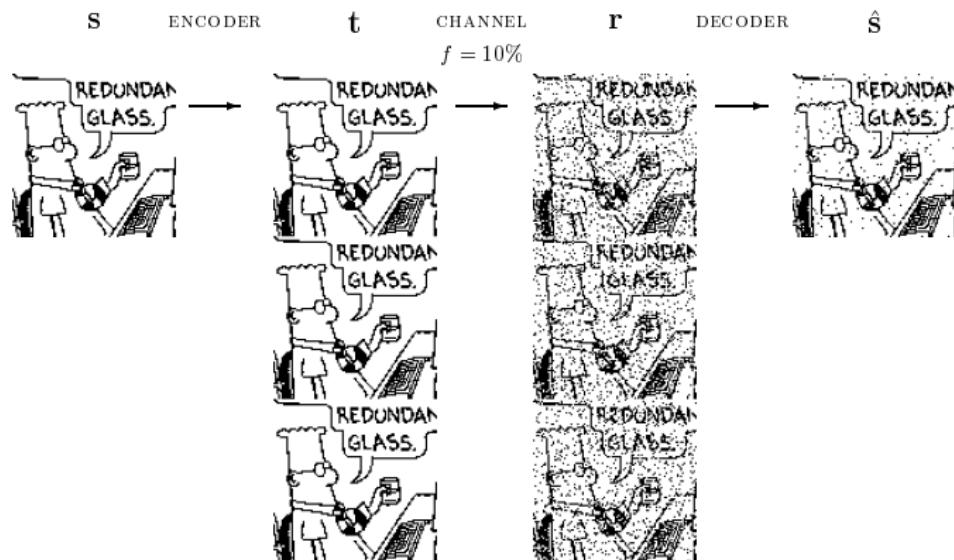
Coding theory: tradeoff between *error correction* and *data rate*.

Performance Comparison

Uncoded case ($f=0.1$)



Triple repetition code



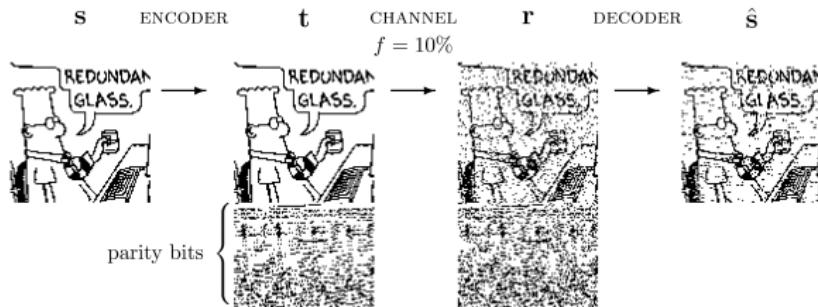
From David J.C. MacKay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.

The 2nd example: (7, 4) Hamming code

Encoding: encode every 4 bit information into 7 bits. (Details are omitted.)

Code rate: $r = 4/7 \approx 0.57$.

Much higher rate but slightly larger P_e .



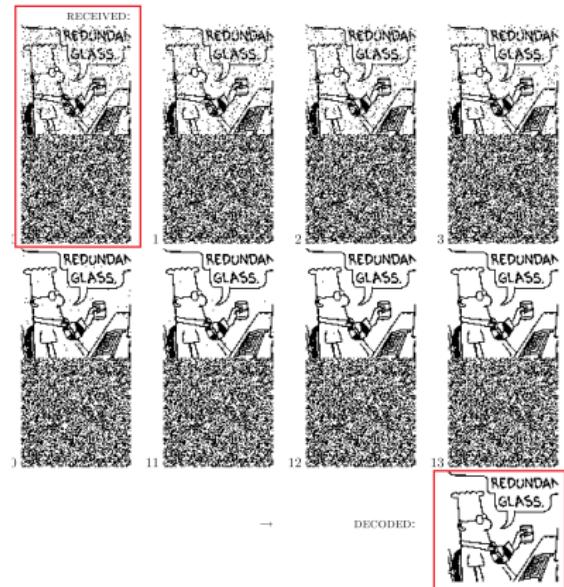
From David J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.

Another example - low-density parity-check code

Details are omitted here. Only simulation is presented

BSC with $p = 7.5\%$.

LDPC (20 000, 10 000) $r = 0.5$



From David J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.

Distance: Definition

Definition 3.2 (Distance)

A distance d on a set \mathcal{X} is a function

$$d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$$

such that for all $x, y, z \in \mathcal{X}$, the following conditions hold:

► Positive definite:

$$d(x, y) \geq 0 \text{ where } "=" \text{ holds iff } x = y.$$

► Symmetry:

$$d(x, y) = d(y, x).$$

► Triangle inequality:

$$d(x, z) \leq d(x, y) + d(y, z).$$

In this course, d is also translation invariant, that is,

$$d(x, y) = d(x + z, y + z).$$

Examples of Commonly Used Distances

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be two vectors of length n , for example,

$$\mathbf{x} = [9, 1, 0], \quad \mathbf{y} = [6, 1, 4] \in \mathbb{R}^3$$

- ▶ ℓ_2 -norm distance: Euclidean distance d_2

$$\begin{aligned} d_2(\mathbf{x}, \mathbf{y}) &= \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \\ &= \sqrt{3^2 + 0^2 + 4^2} = 5. \end{aligned}$$

- ▶ ℓ_1 -norm distance: d_1

$$\begin{aligned} d_1(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^n |x_i - y_i| \\ &= 3 + 0 + 4 = 7. \end{aligned}$$

- ▶ Hamming distance: d_H

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^n \delta_{x_i \neq y_i} \\ &= 1 + 0 + 1 = 2, \end{aligned}$$

where $\delta_{x_i \neq y_i} = 1$ if $x_i \neq y_i$ and $\delta_{x_i \neq y_i} = 0$ if $x_i = y_i$.

Hamming Distance

Definition 3.3 (Hamming Distance)

For $\mathbf{x}, \mathbf{y} \in \mathbb{F}^n$, the Hamming distance is given by

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^n \delta_{x_i \neq y_i} \\ &= |\{i : x_i \neq y_i\}|. \end{aligned}$$

Fact 3.4

Hamming distance is a well defined distance.

To prove this fact, the only non-trivial part is the triangle inequality.

Proof of the Triangle Inequality for Hamming Distance

1. scalar case: $d_H(x_i, z_i) \leq d_H(x_i, y_i) + d_H(y_i, z_i)$:

If $x_i = z_i$, then the equality holds obviously.

If $x_i \neq z_i$, LHS = 1. We have three cases:

$$\left. \begin{array}{l} y_i = x_i \Rightarrow y_i \neq z_i \\ y_i \neq z_i \Rightarrow y_i \neq x_i \\ y_i \neq x_i \text{ and } y_i \neq z_i \end{array} \right\} \Rightarrow \text{RHS} \geq 1.$$

memoryless

2. vector case:

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{z}) &= \sum_{i=1}^n d_H(x_i, z_i) \\ &\leq \sum_{i=1}^n (d_H(x_i, y_i) + d_H(y_i, z_i)) \\ &= d_H(\mathbf{x}, \mathbf{y}) + d_H(\mathbf{y}, \mathbf{z}). \end{aligned}$$

Hamming Distance: Properties

Fact 3.5

Hamming distance is translation invariant:

$$d_H(\mathbf{x}_1, \mathbf{x}_2) = d_H(\mathbf{x}_1 + \mathbf{y}, \mathbf{x}_2 + \mathbf{y}).$$

Definition 3.6 (Weight)

A weight of a vector $\mathbf{x} \in \mathbb{F}_q^n$ is defined as its Hamming distance from the zero vector:

$$\text{wt}(\mathbf{x}) = d_H(\mathbf{x}, 0).$$

Example:

- ▶ $\mathbf{x} = [9, 1, 4], \mathbf{y} = [0, 1, 4] \Rightarrow d_H(\mathbf{x}, \mathbf{y}) = 1.$
- ▶ $\mathbf{x} = [1, 2, 1, 2, 1], \mathbf{y} = [2, 0, 2, 0, 2] \Rightarrow d_H(\mathbf{x}, \mathbf{y}) = 5.$
- ▶ $\mathbf{x} = [0, 1, 0, 1] \Rightarrow \text{wt}(\mathbf{x}) = 2.$

Decoding Techniques

Suppose that a codeword $\mathbf{c} \in \mathcal{C} \subset \mathbb{F}_q^n$ is transmitted and a word \mathbf{y} is received. The decoding function is defined as the mapping

$$\begin{aligned}\mathcal{D} : \mathbb{F}_q^n &\rightarrow \mathcal{C} \\ \mathbf{y} &\mapsto \hat{\mathbf{c}} \in \mathcal{C}.\end{aligned}$$

Popular decoding strategies include

Maximum likelihood decoding:

$$\hat{\mathbf{c}}_{ML} = \mathcal{D}_{ML}(\mathbf{y}) = \arg \max_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{y} \text{ received} | \mathbf{c} \text{ sent}).$$

Minimum distance decoding:

$$\hat{\mathbf{c}}_{MD} = \mathcal{D}_{MD}(\mathbf{y}) = \arg \min_{\mathbf{c} \in \mathcal{C}} d_H(\mathbf{y}, \mathbf{c}).$$

They are equivalent for many channels.

Equivalence Between ML and MD decoding

Theorem 3.7

Consider a memoryless binary symmetric channel (BSC) with cross-over probability $p < 1/2$. Then

$$\hat{\mathbf{c}}_{ML} = \hat{\mathbf{c}}_{MD}.$$

Proof:

$$\Pr(\mathbf{y} \text{ received} | \mathbf{c} \text{ sent}) = \prod_{i=1}^n \Pr(y_i \text{ received} | c_i \text{ sent})$$

$$d_H(\mathbf{y}, \mathbf{c}) \Downarrow \Rightarrow \Pr(\mathbf{y} | \mathbf{c}) \stackrel{\substack{\uparrow \\ \text{error}}}{=} p^{d_H(\mathbf{y}, \mathbf{c})} (1-p)^{n-d_H(\mathbf{y}, \mathbf{c})}$$
$$= (1-p)^n \left(\frac{p}{1-p} \right)^{d_H(\mathbf{y}, \mathbf{c})} < 1.$$

That $p < 1/2$ implies that $p / (1-p) < 1$. Hence, $\Pr(\mathbf{y} \text{ received} | \mathbf{c} \text{ sent})$ is a monotonically decreasing function of $d_H(\mathbf{y}, \mathbf{c})$. The maximum $\Pr(\mathbf{y} | \mathbf{c})$ is achieved when $d_H(\mathbf{y}, \mathbf{c})$ is minimized. ◇

Distance of a Code

Definition 3.8

The **distance** of a code \mathcal{C} is defined as

$$d_H(\mathcal{C}) = \min_{\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{C}, \mathbf{x}_1 \neq \mathbf{x}_2} d_H(\mathbf{x}_1, \mathbf{x}_2).$$

Notation: An (n, M, d) -code:

a code of codeword length n , size M , and distance d .

Example: Consider the binary code

$$\mathcal{C} = \{00000, 00111, 11111\}.$$

It is a binary $(5, 3, 2)$ -code.

Example: Consider the ternary code

$$\mathcal{C} = \{000000, 000111, 111222\}.$$

It is a ternary $(6, 3, 3)$ -code.

Error Detection

Error detector: if the received word $\mathbf{y} \in \mathcal{C}$, let $\hat{\mathbf{c}} = \mathbf{y}$ and claim no error; if $\mathbf{y} \notin \mathcal{C}$, claim that errors happened.

Theorem 3.9

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be an (n, M, d) code. The above error detector detects every pattern of up to $d - 1$ many errors.

Proof:

1. Every pattern of $d - 1$ many errors are detectable. Since at most $d - 1$ many errors happened, $0 < d_H(\mathbf{c}, \mathbf{y}) < d := d(\mathcal{C})$ and $\mathbf{y} \notin \mathcal{C}$. The receiver will claim that errors happened.
2. Exists a pattern of d many errors that is not detectable. By the definition of the code distance, there exist $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}$ s.t. $d_H(\mathbf{c}_1, \mathbf{c}_2) = d$. Suppose that \mathbf{c}_1 is the transmitted codeword and the channel errors happen to be $\mathbf{e} = \mathbf{c}_2 - \mathbf{c}_1$ (d errors happened). Then $\mathbf{y} = \mathbf{c}_2$ is received. As $\mathbf{c}_2 \in \mathcal{C}$, the detector claims that no error happened and set $\hat{\mathbf{c}} = \mathbf{c}_2$.



Error Correction

Theorem 3.10

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be an (n, M, d) code. The minimum distance decoder can correct every pattern of up to $t := \lfloor (d - 1) / 2 \rfloor$ many errors.

Side mark:

$$\begin{array}{c} 0 \curvearrowleft x \curvearrowright x \curvearrowleft 0 \quad 0 \curvearrowleft x \curvearrowright x \curvearrowleft 0 \\ d=4 \quad \quad \quad d=3 \end{array}$$

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor = \begin{cases} \frac{d}{2} - 0.5 & \text{if } d \text{ is odd,} \\ \frac{d}{2} - 1 & \text{if } d \text{ is even.} \end{cases}$$

and

$$\begin{array}{ccc} 2t + 1 \leq d(\mathcal{C}) \leq 2t + 2 & & \\ \text{odd} & & \text{even} \end{array}$$

Examples:

The previous ternary $(6, 3, 3)$ code is exactly 1-error-detecting.

Error Correction: Proof

Proof: Let \mathcal{D} be the minimum distance decoder. Let \mathbf{c} and \mathbf{y} be the transmitted codeword and received word respectively. Let $\hat{\mathbf{c}} = \mathcal{D}_{MD}(\mathbf{y})$.

1. If $d_H(\mathbf{y}, \mathbf{c}) \leq t = \lfloor (d - 1)/2 \rfloor$, then $\hat{\mathbf{c}} = \mathbf{c}$.

Suppose that $\hat{\mathbf{c}} \neq \mathbf{c}$. By the way the decoder \mathcal{D}_{MD} is defined,

$$d_H(\mathbf{y}, \hat{\mathbf{c}}) \leq d_H(\mathbf{y}, \mathbf{c}) \leq t.$$

On the other hand, by the definition of the code distance,

$$d \leq d_H(\mathbf{c}, \hat{\mathbf{c}}) \leq d_H(\mathbf{c}, \mathbf{y}) + d_H(\mathbf{y}, \hat{\mathbf{c}}) \leq 2t \leq d - 1,$$

which is a contradiction.

2. \exists a pair $(\mathbf{c}, \mathbf{y}) \in \mathcal{C} \times \mathbb{F}_q^n$ s.t. $d_H(\mathbf{y}, \mathbf{c}) = t + 1$ and it may happen that $\mathcal{D}_{MD}(\mathbf{y}) \neq \mathbf{c}$.

$d_H(\mathbf{y}, \mathbf{c}) = t + 1 \Rightarrow d_H(\mathbf{y}, \mathbf{c}') > d - (t + 1)$

By the definition of the code distance, $\exists \mathbf{c}, \mathbf{c}' \in \mathcal{C}$ s.t. $d_H(\mathbf{c}, \mathbf{c}') = d$.

W.l.o.g., assume the first d positions of \mathbf{c}, \mathbf{c}' are different. Define \mathbf{y} such that $y_i = c'_i, i = 1, 2, \dots, t + 1$ and $y_i = c_i, i = t + 2, \dots, n$. It is clear that $d_H(\mathbf{y}, \mathbf{c}) = t + 1$ and $d_H(\mathbf{y}, \mathbf{c}') = d - (t + 1)$.

$d_H(\mathbf{y}, \mathbf{c}') = d - (t + 1) \leq t + 1 = d_H(\mathbf{y}, \mathbf{c})$. Hence, it may happen that $\hat{\mathbf{c}} = \mathcal{D}_{MD}(\mathbf{y}) \neq \mathbf{c}$.

$$\therefore d_H(\mathbf{y}, \mathbf{c}) > d_H(\mathbf{y}, \mathbf{c}')$$

Section 4

Linear Codes

- ▶ Definition.
 - ▶ Generator matrices.
 - ▶ Parity-check matrices.
- ▶ Decoding.

Remark: Why linear codes? Low complexity in encoding, decoding, and distance computation.

For the contents relevant to distance, Lin & Xing's book, Chapter 2, should be helpful.

Linear Codes: Definition

Block codes: all codewords are of the same length $\mathcal{C} \subset \mathbb{F}_q^n$.

Definition 4.1 (Linear Codes)

A linear code is a code for which any linear combination of codewords is also a codeword. *(all zeros is always a codeword!)*
That is, let $u, v \in \mathcal{C} \subset \mathbb{F}_q^n$. Then $\lambda u + \mu v \in \mathcal{C}, \forall \lambda, \mu \in \mathbb{F}_q$.

Example of linear codes:

$$\mathcal{C} = \{0000, 0011, 1100, 1111\} \subset \mathbb{F}_2^4.$$

$$\mathcal{C} = \{v \in \mathbb{F}_2^4 : \text{wt}(v) \text{ is even.}\}.$$

Example of nonlinear codes:

$$\mathcal{C} = \{0000, 1100, 1111\}.$$

$$\mathcal{C} = \{v \in \mathbb{F}_3^4 : \text{wt}(v) \text{ is even.}\}.$$

Basis

Definition 4.2 (Basis)

Let $\mathcal{B} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\} \subset \mathbb{F}^n$. It is a basis of a set $\mathcal{C} \subset \mathbb{F}^n$ if it satisfies the following conditions:

► Linear independence property:

For all $\lambda_1, \dots, \lambda_k \in \mathbb{F}$, if $\sum \lambda_i \mathbf{v}_i = \mathbf{0}$, then necessarily
 $\lambda_1 = \dots = \lambda_k = 0$.

► The spanning property:

For every $\mathbf{c} \in \mathcal{C}$, there exist $\lambda_1, \dots, \lambda_k \in \mathbb{F}$ s.t. $\mathbf{c} = \sum_i \lambda_i \mathbf{v}_i$.

$\dim(\mathcal{C}) = k$: the # of vectors in a basis.

The basis \mathcal{B} is not unique in general, but the dimension is.

Example: Let $\mathcal{C} = \{0000, 0011, 1100, 1111\}$.

$\mathcal{B}_1 = \{0011, 1100\}$ is a basis for \mathcal{C} .

$\mathcal{B}_2 = \{0011, 1111\}$ is also a basis for \mathcal{C} .

$\dim(\mathcal{C}) = 2$.

Construct a Basis

Definition 4.3 (Linear Span)

For any subset $\mathcal{V} \subset \mathbb{F}^n$, define $\langle \mathcal{V} \rangle$ as the linear span of \mathcal{V} :

$$\langle \mathcal{V} \rangle = \left\{ \sum \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}, \mathbf{v}_i \in \mathcal{V} \right\}.$$

Construct a basis for a linear code $\mathcal{C} \subset \mathbb{F}^n$:

1. From \mathcal{C} , take a nonzero vector, say \mathbf{v}_1 .
2. Take a nonzero vector, say \mathbf{v}_2 , from $\mathcal{C} - \langle \{\mathbf{v}_1\} \rangle$.
3. Take a nonzero vector, say \mathbf{v}_3 , from $\mathcal{C} - \langle \{\mathbf{v}_1, \mathbf{v}_2\} \rangle$.
4. Continue this process, until $\mathcal{C} - \langle \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\} \rangle = \phi$.
5. Set $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$.

The Size of a Linear Code

Theorem 4.4 **dim = rank = # different symbols**

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be a linear code and $\dim(\mathcal{C}) = k$, then $|\mathcal{C}| = q^k$.

Proof:

1. $\dim(\mathcal{C}) = k \Rightarrow \exists$ a basis $\mathcal{B} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ for \mathcal{C} .

2. $|\mathcal{C}| \leq q^k$: **bound**

Definition of the basis suggests $\mathcal{C} = \langle \mathcal{B} \rangle = \left\{ \sum_{i=1}^k \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}_q \right\}$.

There are q^k many possible linear combinations. Hence, $|\mathcal{C}| \leq q^k$ (repetition may exist).

3. $|\mathcal{C}| = q^k$: **no repetition**

It suffices to show that there is no repetition.

Let $\boldsymbol{\lambda}^{(1)} \neq \boldsymbol{\lambda}^{(2)}$. Let $\mathbf{x}^{(1)} = \sum_{i=1}^k \lambda_i^{(1)} \mathbf{v}_i$ and $\mathbf{x}^{(2)} = \sum_{i=1}^k \lambda_i^{(2)} \mathbf{v}_i$.

Then $\mathbf{x}^{(1)} - \mathbf{x}^{(2)} = \sum_{i=1}^k (\lambda_i^{(1)} - \lambda_i^{(2)}) \mathbf{v}_i \neq \mathbf{0}$ by linear independence of \mathbf{v}_i 's and the fact that $\boldsymbol{\lambda}^{(1)} \neq \boldsymbol{\lambda}^{(2)}$.

There is no repetition in the set $\left\{ \sum_{i=1}^k \lambda_i \mathbf{v}_i : \lambda_i \in \mathbb{F}_q \right\}$. ◊

Generator Matrix

Definition 4.5 (Generator Matrix)

A **generator matrix** G for a linear code $\mathcal{C} \subset \mathbb{F}^n$ is a matrix whose rows form a basis for \mathcal{C} .

For a given (n, k) linear code $\mathcal{C} \subset \mathbb{F}^n$, it can be defined using its generator matrix $G \in \mathbb{F}^{k \times n}$.

The **encoding** function that maps information symbols to a codeword is given by

$S(1 \times k)$: symbol stream
 $E: \mathbb{F}^k \rightarrow \mathcal{C} \subset \mathbb{F}^n$

$G(k \times n)$: gen. matrix $s \mapsto c = sG \in \mathcal{C}$.

Remark:

Encoding of a linear code is efficient: vector-matrix product.

Encoding of a nonlinear code is via a look-up table and hence computationally less efficient.

Examples

Example 1: the $(3, 1)$ repetition code: $\mathbf{G} = [1 \ 1 \ 1]$.

Example 2: the $(7, 4)$ Hamming code.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Example 3: the generator matrix is not unique.

$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ and $\mathbf{G}' = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ generate the same code
 $\mathcal{C} = \{0000, 1010, 0101, 1111\} \subset \mathbb{F}_2^4$.

Dual Code

$S: I \times k$

Definition 4.6 (Dual Code)

$S^\perp: I \times (n-k)$

$C = SG : I \times n$

$C^\perp = S^\perp H : I \times n$

Let $\mathcal{C} \subset \mathbb{F}_q^n$ be a non-empty code. Its **dual code** \mathcal{C}^\perp is defined as

$$\mathcal{C}^\perp = \left\{ v \in \mathbb{F}_q^n : vc^T = \sum_i v_i c_i = 0 \text{ for all } c \in \mathcal{C} \right\}.$$

Lemma 4.7

For any non-empty code $\mathcal{C} \subset \mathbb{F}_q^n$ (linear or nonlinear), its dual code \mathcal{C}^\perp is always linear. !

Proof: Take arbitrary $v_1, v_2 \in \mathcal{C}^\perp$. Then for all $\lambda_1, \lambda_2 \in \mathbb{F}_q$ and for all $c \in \mathcal{C}$,

$$v_1 c^T = 0 \Rightarrow \lambda_1 v_1 c^T = 0$$

$$(\lambda_1 v_1 + \lambda_2 v_2) c^T = 0$$

$$(\lambda_1 v_1 + \lambda_2 v_2) c^T = \lambda_1 v_1 c^T + \lambda_2 v_2 c^T = 0,$$

which implies $\lambda_1 v_1 + \lambda_2 v_2 \in \mathcal{C}^\perp$.



Parity Check Matrix

Definition 4.8 (Parity-Check Matrix)

A parity-check matrix H for a linear code $\mathcal{C} \subset \mathbb{F}_q^n$ is a generator matrix for the dual code \mathcal{C}^\perp .

For a code $\mathcal{C} [n, k]$, it holds that

- $G \in \mathbb{F}_q^{k \times n}$ and $H \in \mathbb{F}_q^{(n-k) \times n}$.
- $H \cdot G^T = 0$. $G \cdot H^T = 0$.

Define a linear code via its parity-check matrix:

gen. & pc. mat.
are defined for
linear codes.

$$\mathcal{C} = \left\{ c \in \mathbb{F}_q^n : \underbrace{cH^T = 0}_{SGH^T = 0} \right\}.$$

Examples

- The $(3, 1)$ repetition code:

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \text{ and } \mathbf{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}.$$

- The $(7, 4)$ Hamming code:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \text{ and } \mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

- A **self-dual** code is a code s.t. $\mathcal{C} = \mathcal{C}^\perp$,

Example: $\mathcal{C} = \{0000, 1010, 0101, 1111\}$, where

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} = \mathbf{H}.$$

Self-dual codes do not exist for vector space \mathbb{R}^n or \mathbb{C}^n .

Relation Between G and H

Consider $\mathcal{C}[n, k] \subset \mathbb{F}_q^n$. Write G and H in systematic forms:

(not necessarily)

► Let $G = [I_k \ A] \in \mathbb{F}_q^{k \times n}$ where $A \in \mathbb{F}_q^{k \times (n-k)}$.

► Let $H = [B \ I_{n-k}] \in \mathbb{F}_q^{(n-k) \times n}$ where $B \in \mathbb{F}_q^{(n-k) \times k}$.

Lemma 4.9

$$GH^T = [I_k \ A] \begin{bmatrix} B^T \\ I_{n-k} \end{bmatrix}$$

It holds that $B = -A^T$.

$$= I_k B^T + A I_{n-k} = 0$$

Proof:

$$\begin{aligned} HG^T &= [B \ I_{n-k}] \begin{bmatrix} I_k \\ A^T \end{bmatrix} = B \cdot I_k + I_{n-k} \cdot A^T \\ &= -A^T + A^T = \mathbf{0} \in \mathbb{F}_q^{(n-k) \times k}. \end{aligned}$$

Systematic form:

Why? Easy to compute H from G , and vice versa.

How? Gaussian-Jordan elimination.

Hamming Weight

Hamming Weight of x : $\text{wt}(x) = |\{i : x_i \neq 0\}| = d(x, 0)$.
non-zero elements

Theorem 4.10

For a linear code \mathcal{C} , $d_H(\mathcal{C}) = \min_{x \in \mathcal{C} \setminus \{\mathbf{0}\}} \text{wt}(x)$.

Proof: $d_H(\mathbf{c}_1, \mathbf{c}_2) = \text{wt}(\mathbf{c}_1 - \mathbf{c}_2) = \text{wt}(\mathbf{c}')$ for some $\mathbf{c}' \in \mathcal{C}$ (by the definition of linear codes).

Notation: $\mathcal{C}[n, k, d]$: n : codeword length; k : dimension; d : distance.

Distance and Parity Check Matrix

Theorem 4.11

$$CH^T = 0$$

Let \mathcal{C} be a linear code defined by the parity-check matrix H . Then that $d(\mathcal{C}) = d$ is equivalent to that

1. Every $d - 1$ columns of H are linearly independent.
2. There exist d linearly dependent columns.

Two Confusing Concepts

Given a matrix H ,

- ▶ **spark:** minimum number of linearly dependent columns
- ▶ **column rank:** maximum number of linearly independent columns.

Theorem 4.11 suggests that $\text{spark}(\mathbf{H}) = d(\mathcal{C})$.

Example 4.12

- ▶ $\mathbf{H} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$: spark $(\mathbf{H}) = 3$ and column rank $(\mathbf{H}) = 2$.
 $(n-k) \times n$
- ▶ $\mathbf{H} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$ is an $n \times (n+1)$ matrix:
spark $(\mathbf{H}) = 2$ and column rank $(\mathbf{H}) = n$.

Application of Theorem 4.11: Binary Hamming Codes

Definition 4.13 (Binary Hamming Codes)

The parity-check matrix of the binary Hamming code

$\mathcal{H}[2^r - 1, 2^r - 1 - r, 3]$ consists of all the nonzero binary vectors (columns) of length r . (Also denoted by \mathcal{H}_r .)

$$k' = n - k - r \Rightarrow k = n - r = 2^r - 1 - r$$

Example 4.14

The $\mathcal{H}_2[3, 1, 3]$ is given by

$$\mathcal{H} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 2 & 3 \end{bmatrix},$$

and the $\mathcal{H}_3[7, 4, 3]$ is given by

$$\mathcal{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{bmatrix}.$$

The Distance of Binary Hamming Codes

Corollary 4.15

The distance of a binary Hamming code is 3, i.e., $d(\mathcal{H}_r) = 3$.

Proof: We apply Theorem 4.11.

- ▶ That there is no zero column implies that the minimum number of linearly dependent columns is at least 2, i.e., $d(\mathcal{C}) = \text{spark}(\mathbf{H}) \geq 2$.
- ▶ In the binary field, that **every two columns are distinct** implies that every two columns are linearly independent. Hence,
 $d(\mathcal{C}) = \text{spark}(\mathbf{H}) \geq 3$.
- ▶ It is easy to see that there exist three columns that are linearly dependent (for example the first three columns). Therefore $d(\mathcal{C}) = 3$.
◊

Corollary 4.16

Binary Hamming codes correct up to one error.

Theorem 4.11: Proof

Proof: Let \mathbf{h}_i be the i^{th} column of \mathbf{H} . $\forall \mathbf{c} \in \mathcal{C}$, let i_1, \dots, i_K be the locations where $c_i \neq 0$. By the definition of parity-check matrix,

$$\mathbf{0} = \sum_{i=1}^n c_i \mathbf{h}_i = \sum_{k=1}^K c_{i_k} \mathbf{h}_{i_k}. \quad \text{d } \mathbf{x} \text{ non-zero components.}$$

$d \times L.D.$

$d(\mathcal{C}) = d \Rightarrow$ Claim 2: $d(\mathcal{C}) = d$ implies that $\exists \mathbf{c} \in \mathcal{C}$ s.t. $\text{wt}(\mathbf{c}) = d$. That is, $\sum_{k=1}^d c_{i_k} \mathbf{h}_{i_k} = \mathbf{0}$ or, $\mathbf{h}_{i_1}, \dots, \mathbf{h}_{i_k}$ are linearly dependent.

$d(\mathcal{C}) = d \Rightarrow$ Claim 1: Suppose not. $\exists \mathbf{h}_{i_1}, \dots, \mathbf{h}_{i_{d-1}}$ are linearly dependent, i.e., $\sum_{k=1}^{d-1} x_{i_k} \mathbf{h}_{i_k} = \mathbf{0}$. Let $\mathbf{x} = [0 \cdots x_{i_1} \cdots x_{i_k} \cdots x_{i_{d-1}} \cdots 0]$. Then $\text{wt}(\mathbf{x}) \leq d - 1$ and $\mathbf{x} \in \mathcal{C}$. Hence $d(\mathcal{C}) \leq d - 1$. A contradiction with $d(\mathcal{C}) = d$.

~~Claims 1&2 $\Rightarrow d(\mathcal{C}) = d$: That every $d - 1$ columns are linearly independent implies no nonzero codeword of weight $d - 1$. That there exists d columns that are linearly dependent means the existence of a codeword of weight d . Hence $d(\mathcal{C}) = \min_{\mathbf{x} \in \mathcal{C} \setminus \{\mathbf{0}\}} \text{wt}(\mathbf{x}) = d$.~~ ◇

Syndrome Vector

Let $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ be a parity-check matrix of a linear code $\mathcal{C}[n, k] \subset \mathbb{F}_q^n$. Suppose that the received word is given by $\mathbf{y} \in \mathbb{F}_q^n$.

Define the syndrome vector

$$\mathbf{CH}^T = \mathbf{0}$$

$$s := \mathbf{yH}^T.$$

$$\mathbf{yH}^T = \mathbf{eH}^T$$

It depends only on the error vector not the transmitted codeword.

In particular, let $\mathbf{y} = \mathbf{x} + \mathbf{e}$ where $\mathbf{x} \in \mathcal{C}$ is the transmitted codeword and $\mathbf{e} \in \mathbb{F}_q^n$ is the error vector introduced by the channel. It holds that

$$s = \mathbf{yH}^T = (\mathbf{x} + \mathbf{e}) \mathbf{H}^T = \mathbf{eH}^T.$$

Syndrome Decoding

MD decoding: Find $\hat{\mathbf{c}} = \arg \min_{\mathbf{c} \in \mathcal{C}} d_H(\mathbf{c}, \mathbf{y})$.

Syndrome decoding:

1. For the received word \mathbf{y} , compute the syndrome vector: $\mathbf{s} := \mathbf{y}\mathbf{H}^T$.
2. Find the error vector \mathbf{e} with the minimum weight: (MD decoding)

$$\hat{\mathbf{e}} = \arg \min_{\mathbf{e}} \text{wt}(\mathbf{e}) \text{ s.t. } \mathbf{s} = \mathbf{e}\mathbf{H}^T. \quad (1)$$

3. Decode \mathbf{y} as $\hat{\mathbf{c}} = \mathbf{y} - \hat{\mathbf{e}}$.

Comments: In general, it is computationally challenging to solve (1). However, all practical codes have particular structures in the parity-check matrix so that the decoding problem can be solved efficiently.

Decoding of Binary Hamming Codes

Take \mathcal{H}_3 (Definition 4.13) as an example.

Assume that $y = [0111111]$. Find the MD decoded codeword $\hat{c} \in \mathcal{C}$.

Since $d(\mathcal{H}_3) = 3$, it corrects up to 1 error.

For any e s.t. $\text{wt}(e) = 1$, let $e_i \neq 0$ for some $i \in [n]$. Then

$$s = eH^T = e_i h_i^T = h_i^T.$$

$$\begin{aligned} S^T &= Hy^T = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned}$$

index > 1 \Rightarrow 1st bit error

In the example, $s = [001]$, $e = [1000000]$ and $\hat{c} = [1111111]$.

Section 5

Coding Bounds

- ▶ Sphere packing (Hamming) bound
- ▶ Sphere covering (Gilbert-Varshamov) bound
- ▶ Singleton bound and MDS codes

The lectures will only cover sphere packing, sphere covering, singleton bounds and relevant contents. Reference: Lin & Xing's book, Chapter 5.

Coding Bounds: Motivation

Consider the Hamming code \mathcal{H}_r :

$$r = 2: [3, 1, 3]$$

$$r = 3: [7, 4, 3]$$

$$r = 4: [15, 11, 3]$$

Questions:

- ▶ Can we do better?
- ▶ What is the best that we can do?

It is **possible** to construct linear codes with parameters

- ▶ $[7, 4, 4]$ over \mathbb{F}_8 . *larger d: less errors,*
- ▶ $[15, 11, 5]$ over \mathbb{F}_{16} . *more corrections.*

Hamming Bound



$$|V| = q^n$$

$$|C| \cdot |V_{\text{ol}}| \leq |V| = q^n$$

Theorem 5.1 (Hamming bound, sphere-packing bound)

For a code of length n and distance d , the number of codewords is upper bounded by

$$M \leq q^n / \left(\sum_{i=0}^t \binom{n}{i} (q-1)^i \right),$$

where $t := \lfloor \frac{d-1}{2} \rfloor$.

$y = y_1 \dots y_n$ $|V_{\text{ol}}| = \sum_{i=0}^t \binom{n}{i} (q-1)^i$
 $x = x_1 \dots x_n$ $/$
 $i \text{ differences}$ $\text{distance of } \pi \cdot y$ $\backslash \text{different values of } y$

Examples

Definition 5.2 (Perfect Codes)

A perfect code is a code that attains the Hamming bound.

- ▶ Binary Hamming code $\mathcal{H}_r [2^r - 1, 2^r - 1 - r, 3]$ is a perfect code.
 $d = 3 \Rightarrow t = \left\lfloor \frac{d-1}{2} \right\rfloor = 1$.
Ball Volume: $\sum_{i=0}^t \binom{n}{i} (q-1)^i = 1 + (2^r - 1) = 2^r$.
Hamming bound: $q^n / \sum_{i=0}^t \binom{n}{i} (q-1)^i = 2^{2^r-1} / 2^r = 2^{2^r-r-1} = 2^k$.
- ▶ Perfect codes are rare (binary Hamming codes & Golay codes).

Hamming Bound: Proof (1)

Define a ball in \mathbb{F}_q^n centered at $\mathbf{x} \in \mathbb{F}_q^n$ with radius t by

$$B(\mathbf{x}, t) = \{\mathbf{y} \in \mathbb{F}_q^n : d(\mathbf{x}, \mathbf{y}) \leq t\}.$$

Step one: the balls $B(\mathbf{c}, t)$, $\mathbf{c} \in \mathcal{C}$, are disjoint.

For all $\mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$, it holds that $B(\mathbf{c}, t) \cap B(\mathbf{c}', t) = \emptyset$.

For a $\mathbf{y} \in B(\mathbf{c}, t)$, then $\mathbf{y} \notin B(\mathbf{c}', t)$ for all $\mathbf{c}' \neq \mathbf{c}$.

By triangle inequality: $d \leq d_H(\mathbf{c}, \mathbf{c}') \leq d_H(\mathbf{c}, \mathbf{y}) + d_H(\mathbf{y}, \mathbf{c}')$.

Then

$$\begin{aligned} d_H(\mathbf{y}, \mathbf{c}') &\geq d - d_H(\mathbf{c}, \mathbf{y}) \geq d - t = d - \left\lfloor \frac{d-1}{2} \right\rfloor \\ &> \left\lfloor \frac{d-1}{2} \right\rfloor = t, \end{aligned}$$

which implies $\mathbf{y} \notin B(\mathbf{c}', t)$.

Hamming Bound: Proof (2)

Step two: Consider the union of these balls.

Clearly $\bigcup_{\mathbf{c} \in \mathcal{C}} B(\mathbf{c}, t) \subset \mathbb{F}_q^n$. Hence,

$$\text{Vol} \left(\bigcup_{\mathbf{c} \in \mathcal{C}} B(\mathbf{c}, t) \right) = \sum_{\mathbf{c} \in \mathcal{C}} \text{Vol}(B(\mathbf{c}, t)) \leq \text{Vol}(\mathbb{F}_q^n) = q^n,$$

where the first equality holds because the balls do not overlap.

The volume of each ball is

$$\text{Vol}(B(\mathbf{c}, t)) = \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Therefore

$$M \text{Vol}(B(\mathbf{c}, t)) \leq q^n \quad \Rightarrow \quad M \leq q^n / \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$



Gilbert-Varshamov Bound

Theorem 5.3 (Gilbert-Varshamov bound, sphere covering bound)

For given code length n and distance d , there exists a code such that

$$q^n / \text{Vol}(d-1) \leq M,$$

where $\text{Vol}(d-1) := \sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i$.

Comment: Different from the sphere packing bound, which holds for all codes, the sphere covering bound claims the existence of a code. That means, some badly designed codes may not satisfy this bound.

Gilbert-Varshamov Bound: Proof

It's proved by construction.

Let $M_0 = \lceil q^n / \text{Vol}(d-1) \rceil > 1$.

It suffices to show that exists a code with M_0 codewords.

Take an arbitrary word $\mathbf{c}_1 \in \mathbb{F}_q^n$.

Since $M_0 > 1$, or $q^n > \text{Vol}(d-1)$, it holds $\mathbb{F}_q^n \setminus B(\mathbf{c}_1, d-1) \neq \emptyset$.

Take an arbitrary word $\mathbf{c}_2 \in \mathbb{F}_q^n \setminus B(\mathbf{c}_1, d-1)$.

It is clear that $d(\mathbf{c}_1, \mathbf{c}_2) \geq d(\mathbf{c}_2 \notin B(\mathbf{c}_1, d-1))$.

Continue this process inductively.

Suppose to obtain codewords $\mathbf{c}_1, \dots, \mathbf{c}_{M_0-1}$ in this way.

Since $\text{Vol} \left(\bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \right) \leq (M_0 - 1) \text{Vol}(d-1) < q^n$,

it holds that $\mathbb{F}_q^n \setminus \bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \neq \emptyset$.

Take an arbitrary $\mathbf{c}_{M_0} \in \mathbb{F}_q^n \setminus \bigcup_{i=1}^{M_0-1} B(\mathbf{c}_i, d-1) \neq \emptyset$.

Let $\mathcal{C} = \{\mathbf{c}_1, \dots, \mathbf{c}_{M_0}\}$.

By construction, $d(\mathbf{c}, \mathbf{c}') > d-1$ for all $\mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$. Hence $d(\mathcal{C}) \geq d$. \diamond

Illustration for Sphere Packing and Covering

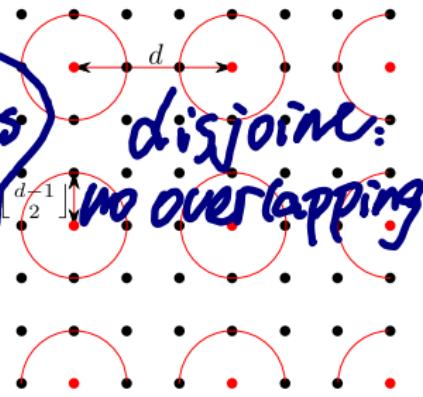
$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$

**Upper bound
by packing**

$$\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i \geq M \geq$$

Sphere Packing

spheres
 $= |C| = M$



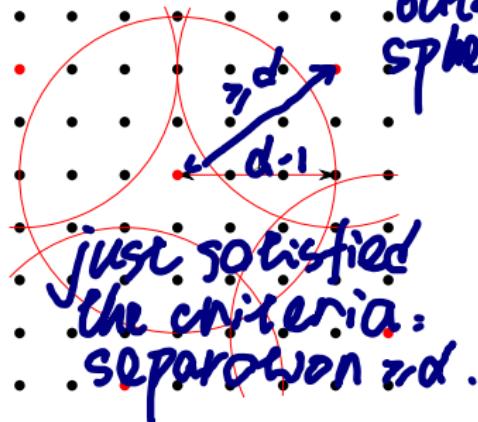
disjoint:
no overlapping

**Lower bound
by covering**

$$\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i$$

Sphere Covering

next center:
outside
spheres.



just satisfied
the criteria:
separation > d.

available space
max nonoverlapping vol. $\geq M \geq$ available space
min nonoverlapping vol.

Singleton Bound and MDS

Theorem 5.4 (Singleton Bound)

The distance of any code $\mathcal{C} \subset \mathbb{F}_q^n$ with M codewords satisfies

$$M \leq q^{n-d+1}.$$

In particular, if the code is linear and $M = q^k$, then

$$d \leq n - k + 1.$$

Definition 5.5 (MDS)

Codes that attain the singleton bound are maximum distance separable (MDS).

Binary Hamming codes $\mathcal{H}_r [2^r - 1, 2^r - 1 - r, 3]$ are **not** MDS in general.

- $d = 3 < n - k + 1 = r + 1$ for all $r \geq 3$.

Singleton Bound: Proof

Proof of the general case:

Let \mathcal{C} be of length n and distance d .

$\forall \mathbf{c} \in \mathcal{C}$, let $\mathbf{c}_{1:n-d+1} \in \mathbb{F}^{n-d+1}$ be the vector containing the first $n - d + 1$ entries of \mathbf{c} , and $\mathbf{c}_{n-d+2:n} \in \mathbb{F}^{d-1}$ be the vector composed of the last $d - 1$ elements of \mathbf{c} .

$\forall \mathbf{c} \neq \mathbf{c}' \in \mathcal{C}$,

$$d \leq d_H(\mathbf{c}, \mathbf{c}') = d_H(\mathbf{c}_{1:n-d+1}, \mathbf{c}'_{1:n-d+1}) + d_H(\mathbf{c}_{n-d+2:n}, \mathbf{c}'_{n-d+2:n}).$$

But $d_H(\mathbf{c}_{n-d+2:n}, \mathbf{c}'_{n-d+2:n}) \leq d - 1$.

Hence, $d_H(\mathbf{c}_{1:n-d+1}, \mathbf{c}'_{1:n-d+1}) \geq d - (d - 1) = 1$.

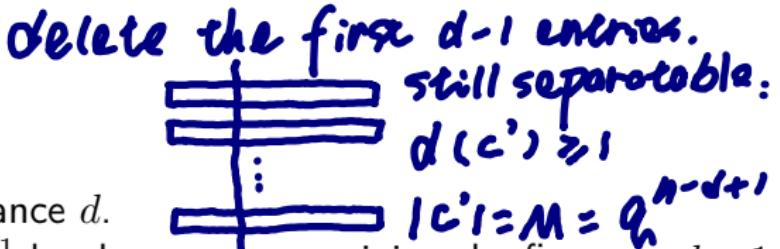
The truncated codewords are all distinct. Hence, $M \leq q^{n-d+1}$. \diamond

Proof for linear codes:

Note that the parity-check matrix $\mathbf{H} \in \mathbb{F}^{(n-k) \times n}$ contains $n - k$ rows.

Every $n - k + 1$ columns must be linearly dependent.

By Theorem 4.11, $d \leq n - k + 1$. \diamond



Dual of MDS Codes

Theorem 5.6

If a linear code \mathcal{C} is MDS, then its dual code \mathcal{C}^\perp is also MDS.

Let the linear code $\mathcal{C}[n, k]$ be MDS.

According to Theorem 5.6, one has

	Parity-check matrix	Generator Matrix	Parameters
\mathcal{C}	$\mathbf{H} \in \mathbb{F}^{(n-k) \times n}$	$\mathbf{G} \in \mathbb{F}^{k \times n}$	$(n, k, n - k + 1)$
\mathcal{C}^\perp	$\mathbf{G} \in \mathbb{F}^{k \times n}$	$\mathbf{H} \in \mathbb{F}^{(n-k) \times n}$	$(n, n - k, k + 1)$

Key for the proof: Theorem 4.11.

If $\mathcal{C}[n, k]$ is MDS, then every set of $n - k$ columns of \mathbf{H} is linearly independent.

$d \leq n - k + 1$
every $(d-1) > n - k$ columns of \mathbf{H} is linearly independent.

Dual of MDS Codes (Theorem 5.6): Proof

Suppose $d(\mathcal{C}^\perp) < k + 1$. Then there exists a nonzero codeword $\mathbf{c} \in \mathcal{C}^\perp$ with at most k nonzero entries and at least $n - k$ zeros. Since permuting the coordinates reserves the codeword weights (i.e., the distance), w.l.o.g., assume that the last $n - k$ coordinates of \mathbf{c} are zeros.

Write the generator matrix of \mathcal{C}^\perp (the parity-check matrix of \mathcal{C}) as $\mathbf{H} = [\mathbf{A}, \mathbf{H}']$, where $\mathbf{A} \in \mathbb{F}^{(n-k) \times k}$ and $\mathbf{H}' \in \mathbb{F}^{(n-k) \times (n-k)}$. By definition of the generator matrix, there exists $\mathbf{s} \in \mathbb{F}^{n-k}$ such that $\mathbf{c} = \mathbf{s}\mathbf{H}$.

As \mathcal{C} is MDS, by Theorem 4.11 the columns of \mathbf{H}' are linearly independent. That is, \mathbf{H}' is invertible. That the last $n - k$ coordinates of \mathbf{c} are zeros implies that $\mathbf{s} = \mathbf{c}_{k+1:n} (\mathbf{H}')^{-1} = \mathbf{0}$. But $\mathbf{s} = \mathbf{0}$ implies $\mathbf{c} = \mathbf{s}\mathbf{H} = \mathbf{0}$ which contradicts the assumption that $\mathbf{c} \neq \mathbf{0}$. Hence, $d(\mathcal{C}^\perp) \geq k + 1$. By the Singleton bound, $d(\mathcal{C}^\perp) = k + 1$. \diamond

Section 6

RS & BCH Codes

- ▶ Reed-Solomon Codes
 - ▶ Definition and properties.
 - ▶ Decoding
- ▶ Cyclic and BCH codes

The contents in this section are significant re-organization and condensation of the materials of many sources, including Lin & Xing's book, Chapters 7 and 8, and Roth's book, Chapters 5, 6 and 8.

Reed-Solomon Codes



Our Heroes: [Irving S. Reed](#) and [Gustave Solomon](#)

Used in

- ▶ Magnetic recording (all computer hard disks use RS codes)
- ▶ Digital versatile disks (CDs, DVDs, etc.)
- ▶ Optical fiber networks (ITU-TG.795)
- ▶ ADSL transceivers (ITU-TG.992.1)
- ▶ Wireless telephony (3G systems, 4G systems)
- ▶ Digital satellite broadcast (ETS 300-421S, ETS 300-429)
- ▶ Deep space exploration (all NASA probes)

RS Codes: Evaluation Mapping

Definition 6.1 (Evaluation Mapping)

Let \mathbb{F}_q be a finite field. Let $n \leq q - 1$ (typically $n = q - 1$).

Let $\mathcal{A} = \{\alpha_1, \dots, \alpha_n\} \subset \mathbb{F}_q$.

For any polynomial $f(x) \in \mathbb{F}_q[x]$, define the evaluation mapping $\text{eval}(f(x))$ that maps f to a vector $\mathbf{c} \in (\mathbb{F}_q)^n$

$$f \mapsto \mathbf{c} = [c_1, \dots, c_n] \text{ where } \underbrace{c_i = f(\alpha_i)}_{\text{eval}}.$$

Example 6.2

$\mathbb{F}_7 = \{0, 1, 2, 3, 4, 5, 6\}$. Choose the primitive element $\alpha = 3$.

Let $\mathcal{A} = \{1, \alpha, \dots, \alpha^5\} = \{1, 3, 2, 6, 4, 5\}$.

$f(x) = 2x + 1$, $\mathbf{c} = \text{eval}(f) = [3, 0, 5, 6, 2, 4]$.

$f(x) = 3x^2 + x + 2$, $\mathbf{c} = \text{eval}(f) = [6, 4, 2, 4, 5, 5]$.

RS Codes: Definition

Definition 6.3 (Reed-Solomon Codes)

Given $\mathcal{A} = \{\alpha_1, \dots, \alpha_n\} \subset \mathbb{F}_q$, an $[n, k]$ q -ary RS code
 $\mathcal{C} = \{\text{eval}(f), 0 \leq \deg f \leq k-1\}$.

The set \mathcal{A} is called a defining set of points of \mathcal{C} .

A common choice of defining set of points of \mathcal{C} is $\mathcal{A} = \{1, \alpha, \dots, \alpha^{q-2}\}$
where α is a primitive element in \mathbb{F}_q .

In this case, $n = q - 1$.

$$\mathcal{C} = \{[f(1), f(\alpha), \dots, f(\alpha^{q-2})] \mid \deg(f) \leq k-1\}$$

$$\mathcal{C} : (q-1, k, q-k)$$

RS Codes: Properties

Theorem 6.4

1. RS codes are linear codes.
2. RS codes are MDS, i.e., The distance of the RS code is $d = n - k + 1$.

Proof:

1. Let $c_1 = \text{eval}(f_1)$ and $c_2 = \text{eval}(f_2)$ where $\deg f_1 \leq k - 1$ and $\deg f_2 \leq k - 1$. Then $\alpha c_1 + \beta c_2 = \text{eval}(g)$ with $g = \alpha f_1 + \beta f_2$. Since $\deg g \leq k - 1$, $\text{eval}(g) \in \mathcal{C}$.
2. A polynomial of degree $\leq k - 1$ can have at most $k - 1$ zeros. Hence, $\forall c \in \mathcal{C} \text{ s.t. } c \neq 0$, $c = \text{eval}(f)$ has weight at least $n - k + 1$. ◇

polynomial deg = $k-1$

⇒ at most $k-1$ zeros for each codeword

⇒ non-zero elements at least $n-k+1$

RS Codes: Conventional Definition

$$[a_0 \cdots a_{k-1}]$$

$$C = SG$$

Theorem 6.7

$$f = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$$

$$[f(1), f(\alpha) \cdots f(\alpha^{n-1})]$$

Let the defining set of points is $\{1, \alpha, \dots, \alpha^{n-1}\}$ with order $(\alpha) = n$
 (typically $n = q - 1$) The generated RS code has generator matrix and
 parity-check matrix given by

$$\begin{aligned} f(1) &= a_0 + a_1 + \cdots + a_{k-1}, \\ f(\alpha) &= a_0 + a_1 \alpha + \cdots + a_{k-1} \alpha^{k-1}, \\ &\vdots \\ f(\alpha^{n-1}) &= a_0 + a_1 \alpha^{n-1} + \cdots + a_{k-1} \alpha^{k-1} \alpha^{n-1} \end{aligned}$$

$G = \left[\begin{array}{cccc} 1 & 1 & \cdots & 1 \\ 1 & \alpha & \cdots & \alpha^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \cdots & \alpha^{(k-1)(n-1)} \end{array} \right]$

$$H = \left[\begin{array}{cccc} 1 & \alpha & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & \cdots & \alpha^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{n-k} & \cdots & \alpha^{(n-k)(n-1)} \end{array} \right]$$

Generator Matrix: Justification

For any $\mathbf{c} \in \mathcal{C}$, there exists a polynomial $f(x) = a_0 + a_1x + \cdots + a_{k-1}x^{k-1}$ s.t. $\mathbf{c} = \text{eval}(f) = [f(1), f(\alpha), \dots, f(\alpha^{n-1})] \in \mathcal{C}$.

Note that $\forall i \in [n]$, $c_i = f(\alpha^i) = \sum_{\ell=0}^{k-1} a_\ell (\alpha^{i-1})^\ell = [a_0, \dots, a_{k-1}] \mathbf{G}_i$ where \mathbf{G}_i is the i -th column of the \mathbf{G} matrix.

One has $\mathcal{C} = \{\mathbf{s}\mathbf{G} : \mathbf{s} \in \mathbb{F}_q^k\}$ and \mathbf{G} is a generator matrix of \mathcal{C} .

Remark: In the definition of the generator matrix (Def. 4.5), the rows of \mathbf{G} are required to be linearly independent. We shall prove it later.

Parity-Check Matrix: Justification

$$G = \begin{bmatrix} 1 & \dots & \alpha^{n-1} \\ 1 & \alpha & \dots & \alpha^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \dots & \alpha^{(k-1)(n-1)} \end{bmatrix} \quad H = \begin{bmatrix} 1 & \alpha & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \dots & \alpha^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{n-k} & \dots & \alpha^{(n-k)(n-1)} \end{bmatrix}$$

Lemma 6.6

The G and H matrices defined in Theorem 6.5 satisfy $GH^T = 0$.

$$A_{ij} = \sum (\text{i}^{\text{th}} \text{ row of } G) (\text{j}^{\text{th}} \text{ row of } H)$$

Proof: Let $A := GH^T \in \mathbb{F}_q^{k \times (n-k)} = \sum_{\ell=1}^n \alpha^{(\ell-1)(i-1)} \cdot \alpha^{j(\ell-1)}$

$\forall i \in [k]$ and $\forall j \in [n-k]$, it holds that

$$= \sum_{\ell=1}^n \alpha^{(\ell-1)(i+j-1)}$$

$$A_{i,j} = \sum_{\ell=1}^n \alpha^{(\ell-1)(i-1)} \alpha^{(\ell-1)j} = \sum_{\ell=1}^n \alpha^{(i+j-1)(\ell-1)}$$

$$\stackrel{(a)}{=} \frac{\alpha^{(i+j-1)n} - 1}{\alpha^{i+j-1} - 1} \stackrel{(b)}{=} 0, \frac{\alpha^{(i+j-1)n} - 1}{\alpha^{i+j-1} - 1} = 0$$

where (a) comes from that $i + j - 1 < n$ and $\alpha^{i+j-1} \neq 1$, and (b) holds because $\alpha^n = 1$.

Row Rank of the \mathbf{G}/\mathbf{H} Matrix

Theorem 6.7

The rows of the \mathbf{G}/\mathbf{H} matrix in Theorem 6.5 are linearly independent.

Proof: It is sufficient to show that any k -column sub-matrix of \mathbf{G} (($n - k$)-column sub-matrix of \mathbf{H}) has full rank.

Note that a k -column sub-matrix of \mathbf{G} is of the form

$$\mathbf{G}' = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha^{i_1} & \alpha^{i_2} & \cdots & \alpha^{i_k} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(k-1)i_1} & \alpha^{(k-1)i_2} & \cdots & \alpha^{(k-1)i_k} \end{bmatrix},$$

which is a Vandermonde matrix (defined and analysed later). A Vandermonde matrix has full rank. Hence the rows of \mathbf{G} are linearly independent. \diamond

Vandermonde Matrix

Definition 6.8 (Vandermonde Matrix)

A Vandermonde matrix $\mathbf{V} \in \mathbb{F}^{n \times n}$ is of the form

$$\mathbf{V} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} \end{bmatrix}.$$

Theorem 6.9

The determinant of a Vandermonde matrix $\mathbf{V} \in \mathbb{F}^{n \times n}$ is

$$|\mathbf{V}| = \prod_{i < j} (\alpha_j - \alpha_i).$$

As a result, if $\alpha_i \neq \alpha_j$, $1 \leq i \neq j \leq n$, then $|\mathbf{V}| \neq 0$ and \mathbf{V} is of full rank.

Determinant: A Recap

Definition 6.10 (Determinant)

$\forall \mathbf{A} \in \mathbb{F}^{n \times n}$, its determinant $|\mathbf{A}|$ is computed via

$$|\mathbf{A}| = \sum_{j=1}^n (-1)^{i+j} a_{i,j} |\mathbf{M}_{i,j}|,$$

where $\mathbf{M}_{i,j}$ is the **minor matrix** obtained by deleting row i and column j from \mathbf{A} .

Lemma 6.11

1. $|\mathbf{AB}| = |\mathbf{A}| |\mathbf{B}|$.
2. If \mathbf{B} results from \mathbf{A} by adding a multiple of one row/column to another row/column, then $|\mathbf{B}| = |\mathbf{A}|$.
3. $|\mathbf{A}| \neq 0 \Leftrightarrow \mathbf{A}$ is of full rank.

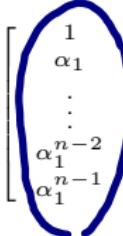
Theorem 6.9: Proof (1)

We prove Theorem 6.9 by using induction.

Recall that

$$\mathbf{V}_n = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_{n-1} & \alpha_n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha_1^{n-2} & \alpha_2^{n-2} & \cdots & \alpha_{n-1}^{n-2} & \alpha_n^{n-2} \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_{n-1}^{n-1} & \alpha_n^{n-1} \end{bmatrix}$$

Let $(\mathbf{V}'_n)_{:,2} = (\mathbf{V}_n)_{:,2} - (\mathbf{V}_n)_{:,1}$, \dots , $(\mathbf{V}'_n)_{:,n} = (\mathbf{V}_n)_{:,n} - (\mathbf{V}_n)_{:,1}$. We obtain

$$\mathbf{V}'_n = \left[\begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ \alpha_1 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha_1^{n-2} & \alpha_2^{n-2} - \alpha_1^{n-2} & \cdots & \alpha_{n-1}^{n-2} - \alpha_1^{n-2} & \alpha_n^{n-2} - \alpha_1^{n-2} \\ \alpha_1^{n-1} & \alpha_2^{n-1} - \alpha_1^{n-1} & \cdots & \alpha_{n-1}^{n-1} - \alpha_1^{n-1} & \alpha_n^{n-1} - \alpha_1^{n-1} \end{array} \right]$$


Theorem 6.9: Proof (2)

Let $(\mathbf{V}_n'')_{n,:} = (\mathbf{V}_n')_{n,:} - \alpha_1 (\mathbf{V}_n')_{n-1,:}, \dots, (\mathbf{V}_n'')_{2,:} = (\mathbf{V}_n')_{2,:} - \alpha_1 (\mathbf{V}_n')_{1,:}$. We obtain

$$\mathbf{V}_n'' = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \alpha_2^{n-2} - \alpha_1 \alpha_2^{n-3} & \cdots & \alpha_{n-1}^{n-2} - \alpha_1 \alpha_{n-1}^{n-3} & \alpha_n^{n-2} - \alpha_1 \alpha_n^{n-3} \\ 0 & \alpha_2^{n-1} - \alpha_1 \alpha_2^{n-2} & \cdots & \alpha_{n-1}^{n-1} - \alpha_1 \alpha_{n-1}^{n-2} & \alpha_n^{n-1} - \alpha_1 \alpha_n^{n-2} \end{bmatrix} \cdot \text{[?]}.$$

$$= \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & \alpha_2 - \alpha_1 & \cdots & \alpha_{n-1} - \alpha_1 & \alpha_n - \alpha_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & (\alpha_2 - \alpha_1) \alpha_2^{n-3} & \cdots & (\alpha_{n-1} - \alpha_1) \alpha_{n-1}^{n-3} & (\alpha_n - \alpha_1) \alpha_n^{n-3} \\ 0 & (\alpha_2 - \alpha_1) \alpha_2^{n-2} & \cdots & (\alpha_{n-1} - \alpha_1) \alpha_{n-1}^{n-2} & (\alpha_n - \alpha_1) \alpha_n^{n-2} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & & & & \\ & \mathbf{V}_{n-1}(\alpha_2, \dots, \alpha_n) & & & \end{bmatrix} \begin{bmatrix} 1 & & & & \\ & \alpha_2 - \alpha_1 & & & \\ & & \ddots & & \\ & & & \alpha_{n-1} - \alpha_1 & \\ & & & & \alpha_n - \alpha_1 \end{bmatrix}$$

Hence $|\mathbf{V}_n| = |\mathbf{V}'_n| = |\mathbf{V}''_n| = |\mathbf{V}_{n-1}| \prod_{j>1} (\alpha_j - \alpha_1)$.



Decoding with Known Error Locations

Let e be the error vector.

Let $\mathcal{I} = \{i : e_i \neq 0\}$ be the set of **error locations**.

$e_{\mathcal{I}}$, $H_{\mathcal{I}}$: sub-vector and sub-matrix of e and H respectively.

If we knew error locations \mathcal{I} :

$$\text{Solve } H_{\mathcal{I}}e_{\mathcal{I}}^T = s^T. (e_{\mathcal{I}}^T = H_{\mathcal{I}}^\dagger s^T)$$

Complexity of pseudo-inverse $H_{\mathcal{I}}^\dagger$: $O(d^3)$.

syndrome

Erasure Correction

Recall the erasure channel model.

Suppose that $\mathbf{c} \in \mathcal{C}$ was transmitted.

Receive $\mathbf{r} = [c_1 \cdots c_{i-1} ? c_{i+1} \cdots c_n]$ (at most $d - 1$ symbols erased).

Decoding: Set the missing symbols to zero, i.e., $\mathbf{r}_{\mathcal{I}} = \mathbf{0}$.

Then $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where $\mathbf{e}_{\mathcal{I}^c} = \mathbf{0}$.

$$\mathbf{s}^T = \mathbf{H}\mathbf{r}^T = \mathbf{H}\mathbf{c}^T + \mathbf{H}\mathbf{e}^T = \mathbf{H}_{\mathcal{I}}\mathbf{e}_{\mathcal{I}}^T.$$

$$\begin{bmatrix} \alpha^{i_1-1} & \alpha^{i_2-1} & \dots & \alpha^{i_s-1} \\ \alpha^{2(i_1-1)} & \alpha^{2(i_2-1)} & \dots & \alpha^{2(i_s-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{s(i_1-1)} & \alpha^{s(i_2-1)} & \dots & \alpha^{s(i_s-1)} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_s \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_s \end{bmatrix}.$$

A Specific Example

Consider a $[7, 4, 4]$ RS code over \mathbb{F}_8 ($\mathbb{F}_2[x]/x^3 + x + 1$).

Let α be a primitive element (a root of $f(x) = x^3 + x + 1$).

$$\mathbb{F}_8: \alpha^3 = \alpha + 1$$

000	001	010	100	011	110	111	101
0	1	α	α^2	α^3	α^4	α^5	α^6

$$\alpha \cdot 1^3 + \alpha \cdot 1^2 + 1 \quad \alpha \cdot \alpha^3 + \alpha \cdot \alpha^2 + \alpha$$

Encoded message $m(x) = \alpha x^3 + \alpha x^2 + x$.

$$c = \text{eval}(m) = [1 \quad \alpha^5 \quad \alpha \quad 1 \quad \alpha^5 \quad \alpha^6 \quad \alpha^5].$$

$$r = [1 \quad \alpha^5 \quad \alpha \quad 1 \quad ? \quad ? \quad \alpha^5].$$

measure \rightarrow error index

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \end{bmatrix}$$

$$cH^T = 0 \Rightarrow \begin{bmatrix} \alpha^4 & \alpha^5 \\ \alpha & \alpha^3 \\ \alpha^5 & \alpha \end{bmatrix} \begin{bmatrix} c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} \alpha \\ 1 \\ \alpha \end{bmatrix} \xrightarrow{\text{matrix inverse}} \begin{bmatrix} c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} \alpha^5 \\ \alpha^6 \end{bmatrix}$$

syndrome known

Error Correction

In previous example:

- ▶ “Error” (erasure) locations are known.
- ▶ The error values are found via matrix inverse.

For error correction:

- ▶ Find error locations
 - ▶ Exhaustive search: complexity $\binom{n}{t} = O(n^t)$.
 - ▶ In practice, \exists methods to find error locations efficiently.
- ▶ Correct errors with given error locations
 - ▶ Methods to avoid matrix inverse.

Efficient Error Correction: Definitions

$$t = \left\lfloor \frac{q-1}{2} \right\rfloor$$

Definitions 6.12 $n = q - 1$

$$d = n - k + 1$$

Syndrome polynomial:

$$S(z) = \sum_{j=0}^{n-k-1} s_j z^j, \text{ where } s = rH^T = eH^T.$$

Error locator polynomial: (information about error locations)

$$L(z) = 0 \text{ iff } z = \alpha^{-i}$$

$$L(z) = \prod_{i \in \mathcal{I}} (1 - \alpha^i z).$$

Error evaluator polynomial: (information about errors)

$$E(z) = L(z) \sum_{i \in \mathcal{I}} \frac{e_i \alpha^i}{(1 - \alpha^i z)} = \sum_{i \in \mathcal{I}} e_i \alpha^i \prod_{j \in \mathcal{I} \setminus \{i\}} (1 - \alpha^j z).$$

$E(z)$ is a single term iff $z = \alpha^{-i}$

Remark: The receiver can compute the syndrome vector and the syndrome polynomial easily.

Information Encoded in $L(z)$ and $E(z)$

- If we know $L(z)$, we can find error locations.

$$\begin{aligned} \blacktriangleright \quad & \begin{cases} L(\alpha^{-k}) = 0 & \text{if } k \in \mathcal{I} \\ L(\alpha^{-k}) \neq 0 & \text{if } k \notin \mathcal{I} \end{cases} \end{aligned}$$

- Error locations can be found by exhaustively computing $L(\alpha^{-k})$,
 $0 \leq k \leq n - 2$.

- $E(z)$ helps find errors e_i , $i \in \mathcal{I}$, without matrix inverse.

- $\forall k \in \mathcal{I}$, $E(\alpha^{-k}) = e_k \alpha^k \prod_{j \neq k} (1 - \alpha^j \alpha^{-k}) \neq 0$.
- $e_k = E(\alpha^{-k}) / (\alpha^k \prod_{j \neq k} (1 - \alpha^j \alpha^{-k}))$.
- or $e_k = -E(\alpha^{-k}) / \frac{d}{dz} L(\alpha^{-k})$, where $\frac{d}{dz} L(z)$ is the derivative of $L(z)$.
- Complexity is reduced from $O(n^3)$ to $O(n^2)$.

Decoding strategy: from $S(z)$ to find $L(z)$ and $E(z)$.

An Example of $L(z)$ and $E(z)$

- ▶ $\mathcal{I} = \{1, 2, 5\}$.
- ▶ $L(z) = (1 - \alpha z)(1 - \alpha^2 z)(1 - \alpha^5 z)$
 - ▶ $L(z) = 0$ if $z = \alpha^{-1}, \alpha^{-2}$, or α^{-5} .
 - ▶ $L(z) \neq 0$ otherwise.
- ▶ $E(z) = e_1 \alpha^1 (1 - \alpha^2 z)(1 - \alpha^5 z)$ T1
 $+ e_2 \alpha^2 (1 - \alpha z)(1 - \alpha^5 z)$ T2
 $+ e_5 \alpha^5 (1 - \alpha z)(1 - \alpha^2 z)$ T3

*Like I,
only one non-zero
term.*

	T1	T2	T3	$E(z)$
$z = \alpha^{-1}$	$\neq 0$	$= 0$	$= 0$	$\neq 0$
$z = \alpha^{-2}$	$= 0$	$\neq 0$	$= 0$	$\neq 0$
$z = \alpha^{-5}$	$= 0$	$= 0$	$\neq 0$	$\neq 0$

Properties of $L(z)$ and $E(z)$

$L(z), E(z)$ do not share common roots.

Let $t = \lfloor \frac{d-1}{2} \rfloor$. $Z = \alpha^{-k}$:

Theorem 6.13 ($\forall k \in I$)

$$\begin{cases} L(z) = 0 \\ E(z) = e_i \alpha^k \prod_{l=I \setminus k} (1 - \alpha^l z) \neq 0 \end{cases}$$

1. $\gcd(L(z), E(z)) = 1$.

2. *The key equation:*

$$E(z) = L(z) S(z) \pmod{z^{d-1}}$$

3. (*Uniqueness*) Let $a(z), b(z) \in \mathbb{F}_q[z]$ be such that $\deg(a(z)) \leq t-1$, $\deg(b(z)) \leq t$, $\gcd(a(z), b(z)) = 1$ and

$$a(z) \equiv S(z) b(z) \pmod{z^{d-1}}.$$

Then $a(z)$ and $b(z)$ are unique up to a constant.

That is, we can treat $a(z) = cE(z)$, $b(z) = cL(z)$, and $E(z)$ and $L(z)$ are generated from an error vector e s.t. $\text{wt}(e) \leq t$.

Decoding Process

1. Compute the syndrome vector and polynomial s and $S(z)$ respectively.

2. Apply Euclidean algorithm to $\underline{z^{d-1}}$ and $\underline{S(z)}$, i.e.,

$$z^{d-1} = q_1(z) S(z) + r_1(z)$$

$$S(z) = q_2(z) r_1(z) + r_2(z)$$

⋮

$$\mathbf{r}_\ell = \gcd(z^{d-1}, S(z))$$

$$r_{\ell-2}(z) = q_\ell(z) r_{\ell-1}(z) + r_\ell(z),$$

where $\deg(r_\ell(z)) \leq t - 1$.

3. By Bézout's Identity (Lem. 1.5), one has

$$\underline{r_\ell(z) = a(z) S(z) + b(z) z^{d-1} \equiv a(z) S(z) \text{ mod } z^{d-1}}.$$

4. Let c be the leading coefficient of the polynomial $a(z)$, i.e., $c^{-1}a(z)$

is a monic polynomial. By Theorem 6.13, set

$$\underline{L(z) = c^{-1}a(z)}, \text{ and } \underline{E(z) = c^{-1}r_\ell(z)}.$$

5. Find the error locations $i \in \mathcal{I}$ from $L(z)$ and the errors e_i from $E(z)$.

$$\hat{\mathbf{c}} = \mathbf{y} - \mathbf{e}.$$

The complexity is highly reduced!

Theorem 6.13, Part 1: Proof

Proof: $L(z)$ has roots α^{-i} , $i \in \mathcal{I}$. None of them is a root of $E(z)$.
 $L(z)$ and $E(z)$ does not share any roots.
 $\gcd(L(z), E(z)) = 1$.

Theorem 6.13, Part 2: Proof

Theorem 6.13 part 2 is a direct consequence of the lemma below.

Lemma 6.14

$$S(z) \equiv \sum_{i \in \mathcal{I}} \frac{e_i \alpha^i}{1 - \alpha^i z} \pmod{z^{d-1}}$$

Proof: As $\mathbf{s} = \mathbf{r}\mathbf{H}^T = \mathbf{e}\mathbf{H}^T$, it follows that

$$\text{Hence, } s_j = \sum_{i=0}^{n-1} e_i \alpha^{i(j+1)} = \sum_{i \in \mathcal{I}} e_i \alpha^{i(j+1)}, \forall 0 \leq j \leq d-2.$$

By the definition of $S(z)$, it holds that

$$\begin{aligned} S(z) &= \sum_{j=0}^{d-2} s_j z^j = \sum_{j=0}^{d-2} \sum_{i \in \mathcal{I}} e_i \alpha^{i(j+1)} z^j \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{d-2} (\alpha^i z)^j \right) \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \left(\sum_{j=0}^{\infty} (\alpha^i z)^j \right) \pmod{z^{d-1}} \\ &= \sum_{i \in \mathcal{I}} e_i \alpha^i \frac{1}{1 - \alpha^i z}. \end{aligned}$$



Theorem 6.13, Part 3: Proof

Proof: To prove the uniqueness, we assume that there exist $(E(z), L(z)) \neq (E'(z), L'(z))$ s.t.

$$E(z) = S(z)L(z) \bmod z^{d-1} \text{ and } E'(z) = S(z)L'(z) \bmod z^{d-1}.$$

It follows that

$$\begin{aligned} E(z)L'(z) &= S(z)L(z)L'(z) \bmod z^{d-1} \\ &= E'(z)L(z) \bmod z^{d-1}. \end{aligned} \tag{2}$$

By assumption, $\deg(E(z)) \leq t - 1$ and $\deg(L'(z)) \leq t$.

It is clear that $\deg(E(z)L'(z)) \leq 2t - 1 \leq d - 2$.

The same is true for $E'(z)L(z)$.

As a result, (2) becomes

$$E(z)L'(z) = E'(z)L(z)$$

Note $\gcd(E(z), L(z)) = 1$. By Lemma 1.12, $E(z) | E'(z)$ and $L(z) | L'(z)$.

Similarly from $\gcd(E'(z), L'(z)) = 1$, $E'(z) | E(z)$ and $L'(z) | L(z)$. Hence, $E(z) = cE'(z)$ and $L(z) = cL'(z)$ for some nonzero $c \in \mathbb{F}_q$. ◇

An Example

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha & \alpha^2 & \alpha^4 & \alpha^5 \\ 1 & \alpha^4 & \alpha & \alpha^3 & \alpha^5 & \alpha^2 & \alpha^6 \\ 1 & \alpha^5 & \alpha^4 & \alpha^2 & \alpha^6 & \alpha & \alpha^3 \end{bmatrix}$$
$$S = yH^T = eH^T = [\alpha^3, \alpha^4, \alpha^4, 0]$$
$$S(z) = \alpha^3 z^3 + \alpha^4 z^4 + \alpha^4 z$$

$$d = n - k + 1 = 5$$

Example: Consider the $[7, 3]$ RS code over \mathbb{F}_8 (\mathbb{F}_8 is given as follows).

0	1	α	α^2	α^3	α^4	α^5	α^6
000	001	010	100	011	110	111	101

Let the received signal be $y = [\alpha^3, \alpha, 1, \alpha^2, 0, \alpha^3, 1]$. Find \hat{c} .

Solutions to the Example

1. **Parameters:** $n - k = 4$, $d = 5$ ($t = 2$), and $\mathbf{H} \in \mathbb{F}_8^{4 \times 7}$.

$$\mathbf{H} = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \\ 1 & \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 \end{bmatrix}.$$

2. **Syndromes:** $\mathbf{s} = \mathbf{y}\mathbf{H}^T = [\alpha^3, \alpha^4, \alpha^4, 0]$.

$$S(z) = \alpha^4 z^2 + \alpha^4 z + \alpha^3.$$

3. **Key polynomials:** apply Euclidean algorithm to z^4 and $S(z)$.

3.1 $z^4 = (\alpha^3 z^2 + \alpha^3 z + \alpha^5) S(z) + (z + \alpha)$.

3.2 $L'(z) = \alpha^3 z^2 + \alpha^3 z + \alpha^5$. $E'(z) = z + \alpha$.

3.3 $L(z) = \alpha^5 z^2 + \alpha^5 z + 1$. $E(z) = \alpha^2 z + \alpha^3$.

4. **Find $\hat{\mathbf{c}}$:**

4.1 Plug $1, \alpha^{-1}, \dots$ into $L(z)$. $L(\alpha^{-2}) = L(\alpha^{-3}) = 0$.

4.2 According $E(z)$, we have $e_2 = \alpha^3$ and $e_3 = \alpha^6$.

4.3 $\hat{\mathbf{c}} = \mathbf{y} - \mathbf{e} = \mathbf{y} + \mathbf{e} = [\alpha^3, \alpha, \alpha, 1, 0, \alpha^3, 1]$.

Towards Cyclic and BCH Codes

Have seen

- ▶ Binary Hamming codes: $d = 3$.
- ▶ Reed-Solomon codes: MDS ($d = n - k + 1$) and requires large fields (typically $q = n + 1$).

Will introduce cyclic codes

- ▶ Reed-Solomon codes are a special case of cyclic codes.
- ▶ BCH codes as another special case.
 - ▶ Systematic way to construct binary codes with large distance.

Cyclic Codes

Definition 6.15

An $[n, k]$ linear code is **cyclic**

if for every codeword $c = c_0c_1 \cdots c_{n-2}c_{n-1}$,

the right cyclic shift of c , $c_{n-1}c_0c_1 \cdots c_{n-2}$, is also a codeword.

Example: The $\mathcal{H}[7, 3]$ has the parity-check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Its dual code \mathcal{H}_3^\perp (view \mathbf{H} as the generator matrix) is cyclic.

(The codewords are 1011100, 0101110, 0010111, 1110010, 1001011, 0111001, 1100101, 0000000.)

Generating Function

Definition 6.16

The generating function of a codeword $\mathbf{c} = [c_0 \cdots c_{n-1}]$ is
 $c(x) = c_0 + c_1x + \cdots + c_{n-1}x^{n-1}$.

It will be convenient to use $c(x)$ to represent a codeword \mathbf{c} .

The right cyclic shift $\mathbf{c} = (c_0, \dots, c_{n-1}) \mapsto \mathbf{c}' = (c_{n-1}, c_0, \dots, c_{n-2})$
can be obtained by $c'(x) = x \cdot c(x) \bmod x^n - 1$ as

$$\begin{aligned} x \cdot c(x) &= c_0x + c_1x^2 + \cdots + c_{n-2}x^{n-1} + c_{n-1}x^n \\ &= c_{n-1} + c_0x + c_1x^2 + \cdots + c_{n-2}x^{n-1} \bmod x^n - 1. \end{aligned}$$

Lemma 6.17

Let $c(x) \in \mathcal{C}$. For an arbitrary $u(x)$, $u(x)c(x) \bmod x^n - 1$ is in \mathcal{C} .

Generator Polynomial

Theorem 6.18

generator polynomial

For a cyclic code \mathcal{C} , \exists a **unique** monic polynomial $g(x)$ s.t.

for all $c(x) \in \mathcal{C}$, $c(x) = u(x)g(x)$ for some $u(x)$.

Proof:

Let $g(x) \in \mathcal{C}$ be the nonzero polynomial **of least degree**.

Since \mathcal{C} is linear, w.l.o.g., assume that $g(x)$ is monic.

Then $\forall c(x) \in \mathcal{C}$, write $c(x) = u(x)g(x) + r(x)$.

By definition of cyclic codes, $u(x)g(x) \in \mathcal{C}$.

Hence, $r(x) \in \mathcal{C}$ by linearity of \mathcal{C} .

But $\deg(r(x)) < \deg(g(x))$, which implies $r(x) = 0$.

The uniqueness of $g(x)$ can be proved by contradiction. Suppose that there are two *monic* polynomials $g_1(x) \neq g_2(x)$ of the same degree that both generate \mathcal{C} . Then $g_1(x) - g_2(x) \in \mathcal{C}$ and $\deg(g_1 - g_2) < \deg(g_1)$, which forces $g_1(x) - g_2(x) = 0$. \diamond

Properties of the Generator Polynomial

Corollary 6.19

$$g(x) | x^n - 1.$$

Proof: Write $x^n - 1 = q(x)g(x) + r(x)$.

Take "mod $x^n - 1$ " on both sides.

$$0 = x^n - 1 \bmod x^n - 1 \in \mathcal{C}. \quad q(x)g(x) \bmod x^n - 1 \in \mathcal{C}(x).$$

$$\text{Hence } r(x) \bmod x^n - 1 \in \mathcal{C} \Rightarrow r(x) \in \mathcal{C} \Rightarrow r(x) = 0. \quad \diamond$$

Remark: Let $n = q^m - 1$.

We know how to factor $x^n - 1$ in terms of minimal polynomials.

$g(x)$ must be a product of minimal polynomials.

Generator Matrices of Cyclic Codes

Theorem 6.20

The generator matrix of a cyclic code $\mathcal{C} [n, k]$:

$$G = \left[\begin{array}{c} g(x) \\ xg(x) \\ \vdots \\ x^{k-1}g(x) \end{array} \right] = \left[\begin{array}{cccc} g_0 & g_1 & \cdots & g_{n-k} \\ g_0 & g_1 & \cdots & g_{n-k} \\ \ddots & \ddots & \ddots & \ddots \\ g_0 & g_1 & \cdots & g_{n-k} \end{array} \right].$$

Observations:

- ▶ $\deg(g(x)) = n - k$.
- ▶ Easy for implementation: can be implemented by using flip-flops.

Parity-Check Matrices of Cyclic Codes

$$\deg(g(x)) = n-k$$
$$\deg(h(x)) = k$$

Recall $g(x) | x^n - 1$. Define $h(x)$ such that $g(x)h(x) = x^n - 1$. Then $h(x)$ is a *monic* polynomial with *degree* k .

Write $h(x) = \sum_{i=0}^k a_i x^i$.

Definition 6.21

The reciprocal polynomial $h_R(x)$ of $h(x)$ is given by

$$h_R(x) = a_k + a_{k-1}x + \cdots + a_0x^k = x^k h(1/x).$$

Example: $h(x) = 1 + x^2 + x^3 \Rightarrow h_R(x) = 1 + x + x^3$.

$$\begin{array}{cccc} 1 & 0 & 1 & 1 \\ \hline \end{array}$$

$$\begin{array}{cccc} 1 & 1 & 0 & 1 \\ \hline \end{array}$$

Parity-Check Matrix

Theorem 6.22

The parity-check matrix of the cyclic code $\mathcal{C} [n, k]$ is

$$\mathbf{H} = \left[\begin{array}{c} h_R(x) \\ xh_R(x) \\ \vdots \\ x^{n-k-1}h_R(x) \end{array} \right] = \left[\begin{array}{cccc} h_k & h_{k-1} & \cdots & h_0 \\ h_k & h_{k-1} & \cdots & h_0 \\ \ddots & \ddots & \ddots & \ddots \\ h_k & h_{k-1} & \cdots & h_0 \end{array} \right].$$

Corollary 6.23

The dual of a cyclic code, \mathcal{C}^\perp , is also cyclic.

$h_0^{-1}h_R(x)$ is the generator polynomial of \mathcal{C}^\perp .

Theorem 6.22: Proof

By assumption, $x^n - 1 = g(x)h(x)$. Note that

$$\begin{aligned}g(x)h(x) &= \left(\sum_{i=0}^{n-k} g_i x^i\right) \left(\sum_{i=0}^k h_i x^i\right) \\&= \sum_{i=0}^n \left(\sum_{\ell=0}^i g_\ell h_{i-\ell}\right) x^i = \sum_{i=0}^n a_i x^i,\end{aligned}$$

where $a_0 = g_0 h_0 = -1$, $a_n = g_{n-k} h_k = 1 \cdot 1 = 1$, and

$$a_i = \sum_{\ell=0}^i g_\ell h_{i-\ell} = 0, \quad 1 \leq i \leq n-1.$$

Let $\mathbf{A} = \mathbf{G}\mathbf{H}^T$ with

$$\mathbf{A}_{i,j} = [\underbrace{0, \dots, 0}_{i-1}, g_0, \dots, g_{n-k}, 0, \dots, 0] \cdot [\underbrace{0, \dots, 0}_{j-1}, h_k, \dots, h_0, 0, \dots, 0]^T.$$

It can be verified that $A_{1,1} = a_k$, $A_{1,2} = a_{k+1}$, \dots , and

$$\mathbf{A} = \mathbf{G}\mathbf{H}^T = \begin{bmatrix} a_k & a_{k+1} & \cdots & a_{n-1} \\ a_{k-1} & a_k & \cdots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_{n-k} \end{bmatrix} = \mathbf{0} \in \mathbb{F}^{k \times (n-k)} \quad \diamond$$

Cyclic Codes: An Example

To construct a cyclic code on \mathbb{F}_q , we realize that

- ▶ $M^{(i)}(x) \in \mathbb{F}_q[x]$
- ▶ $M^{(i)}(x) | x^{q^m-1} - 1.$

Definition 6.24

A **BCH code** over \mathbb{F}_q of length $n = q^m - 1$ is the cyclic code generated by
$$g(x) = \text{lcm}(M^{(a)}(x), \dots, M^{(a+\delta-2)}(x))$$

for some integer a . (The code is called **narrow-sense** if $a = 1$.)

Lemma 6.25

A BCH code defined in Definition 6.24 has $d \geq \delta$.

δ is referred to the designed distance.

Distance of BCH Codes: Proof of Lemma 6.25

Let α be the primitive element in \mathbb{F}_{q^m} . By construction, $\alpha^a, \dots, \alpha^{a+\delta-2}$ are roots of the generator polynomial $g(x)$.

That is, $\forall c \in \mathcal{C}$, the generating function $c(x)$ satisfies $c(\alpha^i) = 0$, $a \leq i \leq a + \delta - 2$. In matrix format,

$$\begin{bmatrix} 1 & \alpha^a & \dots & \alpha^{a(n-1)} \\ 1 & \alpha^{a+1} & \dots & \alpha^{(a+1)(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{a+\delta-2} & \dots & \alpha^{(a+\delta-1)(n-1)} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \mathbf{0} \quad (3)$$

Any $(\delta - 1)$ -column submatrix is a Vandermonde matrix and hence of full rank. This implies $d \geq \delta$. \diamond

Remark: The matrix in (3) is in $\mathbb{F}_{q^m}^{(\delta-1) \times n}$ while the vector $c \in \mathcal{C} \subset \mathbb{F}_q^n$. Hence the matrix is not a parity-check matrix when $m > 1$.

Example: Reed-Solomon Codes

Recall that a RS-code $\mathcal{C} [n, k, n - k + 1]$ is built on \mathbb{F}_q with typically $n = q - 1$.

Compare the parity-check matrix of a RS-code (Theorem 6.5) and Equation (3). It is clear that a RS-code is a special case of a BCH code with $m = 1$.

In particular, suppose that we are asked to build a BCH code over \mathbb{F}_q with $n = q - 1$ and $d \geq \delta = n - k + 1$. We find $M^{(i)}(x) \subset \mathbb{F}_q[x]$, $1 \leq i \leq 1 + \delta - 2 = n - k$. Since $M^{(i)}(x) \subset \mathbb{F}_q[x]$, it follows that $M^{(i)}(x) = x - \alpha^i$. Hence $g(x) = \prod_{i=1}^{n-k} (x - \alpha^i)$ and the generator matrix can be constructed (in a different form of that in Theorem 6.5) and good for implementation). Its parity-check matrix is given by the matrix in Equation (3). RS decoder can be directly applied for decoding.

Example: Binary BCH Codes

We have learned binary Hamming codes. The distance is always 3. The question is how to construct a binary code with large distance.

For example, how to construct a binary code of length 15 and $d \geq 5$?

1. For binary codes, use \mathbb{F}_2 . $n = 15 = 2^4 - 1$ hence $m = 4$.
2. $\delta = 5$ implies $g(x) = \text{lcm}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x))$.
3. The relevant cyclotomic cosets of 2 modulo 15 include

$\mathcal{C}_1 = \{1, 2, 4, 8\}$ and $\mathcal{C}_3 = \{3, 6, 9, 12\}$. Hence

$M^{(1)}(x) = \prod_{i \in \mathcal{C}_1} (x - \alpha^i) = M^{(2)}(x) = M^{(4)}(x)$ and
 $M^{(3)}(x) = \prod_{i \in \mathcal{C}_3} (x - \alpha^i)$. Furthermore,

$$g(x) = M^{(1)}(x) \cdot M^{(3)}(x).$$

4. Find the generator matrix and parity-check matrix according to Theorems 6.20 and 6.22 respectively.

From Hamming to BCH

Example: A binary code of length 15 and $d \geq 5$?

$$\begin{aligned}g(x) &= \text{lcm}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)) \\&= \text{lcm}(M^{(1)}(x), M^{(3)}(x)) \\&= M^{(1)}(x) \times M^{(3)}(x).\end{aligned}$$

RS codes are special cases of BCH codes ($m = 1$).

Have learned [7, 4, 3] Hamming code.

$$\boldsymbol{H} = \left[\begin{array}{ccccccc} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{array} \right]$$

Another view:

Let α be a primitive element of \mathbb{F}_8 that satisfies $\alpha^3 = \alpha + 1$.

The parity check matrix can be written as

$$\left[1 \quad \alpha \quad \alpha^2 \quad \alpha^3 \quad \alpha^4 \quad \alpha^5 \quad \alpha^6 \right].$$

(Column vectors in $\mathbb{F}_2^3 \mapsto$ Elements in \mathbb{F}_8)

From Hamming to BCH: Larger Distance

Binary BCH codes with $d \geq 5$:

$$\begin{aligned}g(x) &= \text{lcm}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)) \\&= \text{lcm}(M^{(1)}(x), M^{(3)}(x)).\end{aligned}$$

It holds that

$$\left[\begin{array}{ccccccc} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha & \alpha^3 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \\ 1 & \alpha^4 & \alpha & \alpha^5 & \alpha^2 & \alpha^6 & \alpha^3 \end{array} \right] \mathbf{c} = \mathbf{0}$$

$$M^{(1)}(x) = M^{(3)}(x) = M^{(4)}(x) = 0$$

But $c(\alpha) = 0 \Rightarrow \begin{cases} c(\alpha^2) = c(\alpha) = 0 \\ c(\alpha^4) = c(\alpha)^4 = 0 \end{cases}$

$$\mathbf{H} = \left[\begin{array}{ccccccc} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 \\ 1 & \alpha^3 & \alpha^6 & \alpha^2 & \alpha^5 & \alpha & \alpha^4 \end{array} \right]$$

Eventually, we get a $[7, 1, 7]$ code $\mathcal{C} = \{0000000, 1111111\}$.