Coding Theory

CO 331

Alfred Menezes

LaTeXed by Sibelius Peng

Preface

Disclaimer Much of the information on this set of notes is transcribed directly/indirectly from the lectures of CO 331 during Winter 2021 as well as other related resources. I do not make any warranties about the completeness, reliability and accuracy of this set of notes. Use at your own risk.

Note that the notes is not complete, but should contain the main results. For a complete version, please refer to

- old notes which is not well-structured; or
- Cameron's notes, which I have corrected a number of typos.

For any questions, send me an email via https://notes.sibeliusp.com/contact.

You can find my notes for other courses on https://notes.sibeliusp.com/.

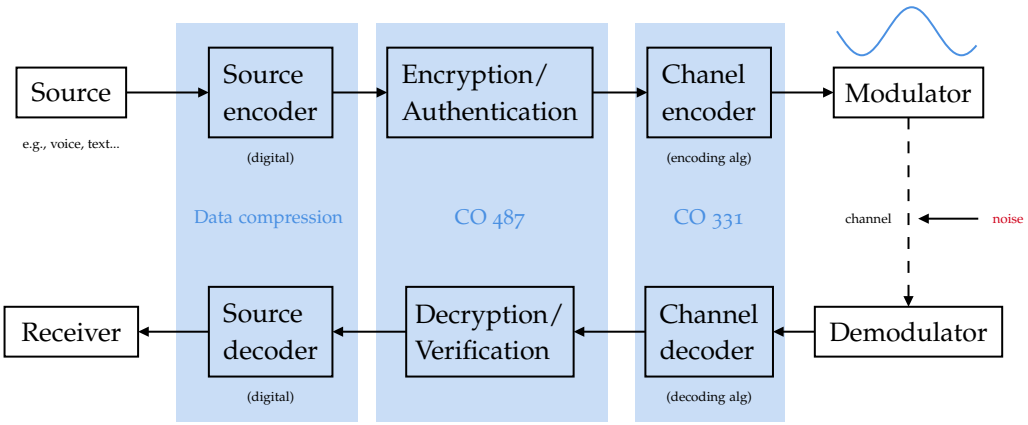
Sibelius Peng

Contents

Introduction

Coding theory is about clever ways of adding redundancy to messages to allow (efficient) error detection and error correction.

Here is our communication model:



Example: Parity Code

Encoding algorithm Add a 0 bit to the (binary) msg m if the number of 1's in m is even; else add a 1 bit.

Decoding algorithm If the number of 1's in a received msg r is even, then accept r; else declare that an error has occurred.

Example: Replication Code

Source msgs	Codeword	# err/codeword (always) detected	# err/codeword (always) corrected *	Information rate
0 1	0 1	0	0	1
0 1	00 11	1	0	$\frac{1}{2}$
0 1	000 111	2	1	$\frac{1}{3}$
0 1	0000 1111	3	1	$\frac{1}{4}$
0 1	00000 11111	4	2	$\frac{1}{5}$

$\xrightarrow{\text{encoding algorithm}}$

*: using "nearest neighbour decoding"

Goal of Coding Theory

Design codes so that:

1. High information rate
2. High error-correcting capability
3. Efficient encoding & decoding algorithms

Course Overview

This course deals with *algebraic methods* for designing good (block) codes. The focus is on error correction (not on error detection). These codes are used in wireless communications, space probes, CD/DVD players, storage, QR codes, etc.

Some modern stuff are not covered: Turbo codes, LDPC codes, Raptor codes, ... Their math theories are not so elegant as algebraic codes.

The big picture

Coding theory in its broadest sense deals with techniques for the *efficient*, *secure* and *reliable* transmission of data over communication channels that may be subject to *non-malicious errors* (noise) and *adversarial intrusion*. The latter includes passive intrusion (eavesdropping) and active intrusion (injection/deletion/modification).

1

Fundamentals

1.1 Basic Definitions and Concepts

alphabet

An **alphabet** A is a finite set of $q \geq 2$ symbols.

word

A **word** is a finite sequence of symbols from A (also: vector, tuple).

length

The **length** of a word is the number of symbols it has.

code

A **code** C over A is a set of words (of size ≥ 2).

codeword

A **codeword** is a word in the code C.

block code

A **block code** is a code in which all codewords have the same length.

A **block code of length** n **containing** M **codewords over** A is a subset $C \subseteq A^n$ with $|C| = M$. C is called an $[n, M]$-code over A.

Example:

$A = \{0,1\}$. $C = \{00000, 11100, 00111, 10101\}$ is a $[5,4]$-code over $\{0,1\}$.

Messages		Codewords
00	$\rightarrow$	00000
10	$\rightarrow$	11100
01	$\rightarrow$	00111
11	$\rightarrow$	10101

↑ Encoding of messages (1-1 map)

Assumptions about the communications channel

(1) The channel only transmits symbols from A ("hard decision decoding").

(2) No symbols are deleted, added, interchanged or transposed during transmission.

(3) The channel is a q-symmetric channel:

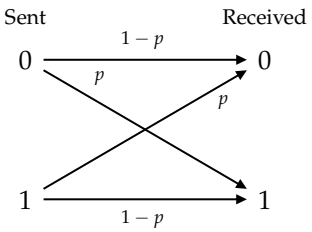
Let $A = \{a_1, \ldots, a_q\}$. Let X_i = the i^{th} symbol sent. Let Y_i = the i^{th} symbol received. Then for all $i \geq 1$, and all $i \leq j, k \leq q$,

$$\Pr(Y_i = a_j | X_i = a_k) = \begin{cases} 1-p, & \text{if } j = k \\ \frac{p}{q-1}, & \text{if } j \neq k. \end{cases}$$

p is called the **symbol error probability** of the channel $(0 \leq p \leq 1)$.

Binary Symmetric Channel (BSC)

A 2-symmetric channel is called a binary symmetric channel.



For a BSC:

1. If $p = 0$, the channel is *perfect*.
2. If $p = 1/2$, the channel is *useless*.
3. If $1/2 < p \leq 1$, then flipping all received bits converts the channel to a BSC with $0 \leq p < 1/2$.
4. Henceforth, we will assume that $0 < p < 1/2$ for a BSC.

Exercise:

For a q-symmetric channel, show that one can take $0 < p < \frac{q-1}{q}$ WLOG.

One can first consider the case $q = 3$.

information rate

The **information rate** (or rate) R of an $[n, M]$-code C over A is $R = \frac{\log_q M}{n}$.

If C encodes messages that are k-tuples over A (so $M = |A^k| = q^k$), then $R = \frac{k}{n}$.

Note:
$0 \leq R \leq 1$. Ideally, R should be close to 1.

Example:
The rate of the binary code $C = \{00000, 11100, 00111, 10101\}$ is $R = \frac{2}{5}$.

Hamming distance

The **Hamming distance** (or distance) between two n-tuples over A is the number of coordinate positions in which they differ.

The Hamming distance (or distance) of an $[n, M]$-code C is $d(C) = \min\{d(x, y) : x, y \in C, x \neq y\}$.

Example:
The distance of $C = \{00000, 11100, 00111, 10101\}$ is $d(C) = 2$.

Theorem 1.1: properties of Hamming distance

For all $x, y, z \in A^n$,

1. $d(x, y) \geq 0$, with $d(x, y) = 0$ iff $x = y$.
2. $d(x, y) = d(y, x)$.
3. $d(x, y) + d(y, z) \geq d(x, y)$ ($\triangle$ inequality).

1.2 Decoding Strategy

Example:
Let $C = \{00000, 11100, 00111, 10101\}$. C is a $[5, 4]$-code over $\{0, 1\}$ (a binary code).

Error Detection If C is used for error detection only, the strategy is the following: A received word $r \in A^n$ is accepted if and only if $r \in C$.

Error Correction Let C be an $[n, M]$-code over A with distance d. Suppose $c \in C$ is transmitted, and $r \in A^n$ is received. The (channel) decoder must decide one of the following:

(i) No errors have occurred; *accept* r.

(ii) Errors have occurred; *correct*[1] (*decode*) r to a codeword $c \in C$?

(iii) Errors have occurred; *no correction is possible*.

Nearest Neighbour Decoding

(i) Incomplete Maximum Likelihood Decoding (IMLD):

If there is a unique codeword $c \in C$ such that $d(r, c)$ is minimum, then correct r to c. If no such c exists, then report that errors have occurred, but correction is not possible (ask for retransmission, or disregard information).

(ii) Complete Maximum Likelihood Decoding (CMLD):

[1] Error correction does not guarantee that the channel decoder always makes the correct decision. For example, 00000 $\xrightarrow{\text{transmit}}$ 11100 which is accepted.

Same as IMLD, except that if there are two or more $c \in C$ for which $d(r,c)$ is minimum, correct r to an arbitrary one of these.

Is IMLD a reasonable strategy?

Theorem 1.2

IMLD chooses the codeword c for which the conditional probability

$$P(r|c) = P(r \text{ is received}|c \text{ is sent})$$

is largest.

Proof:
Suppose $c_1, c_2 \in C$ with $d(c_1,r) = d_1$ and $d(c_2,r) = d_2$. Suppose $d_1 > d_2$.

Now

$$P(r|c_1) = (1-p)^{n-d_1}\left(\frac{p}{q-1}\right)^{d_1}$$

and

$$P(r|c_2) = (1-p)^{n-d_2}\left(\frac{p}{q-1}\right)^{d_2}$$

So,

$$\frac{P(r|c_1)}{P(r|c_2)} = (1-p)^{d_2-d_1}\left(\frac{p}{q-1}\right)^{d_1-d_2} = \left(\frac{p}{(1-p)(q-1)}\right)^{d_1-d_2}$$

Recall

$$p < \frac{q-1}{q} \implies pq < q-1 \implies 0 < q-pq-1$$

$$\implies p < p+q-pq-1 \implies p < (1-p)(q-1) \implies \frac{p}{(1-p)(q-1)} < 1$$

Hence

$$\frac{P(r|c_1)}{P(r|c_2)} < 1$$

and so

$$P(r|c_1) < P(r|c_2)$$

and the result follows. □

Minimum Error Probability Decoding (MED)

An *ideal strategy* would be to correct r to a codeword $c \in C$ for which $P(c|r) = P(r \text{ is received}|c \text{ is sent})$ is largest. This is MED.

Example: (IMLD/CMLD) is not the same as MED

Consider $C = \{\underset{\uparrow c_1}{000}, \underset{\uparrow c_2}{111}\}$. Suppose $P(c_1) = 0.1$ and $P(c_2) = 0.9$. Suppose $p = \frac{1}{4}$ (for a BSC).

Suppose $r = 100$ is the received word. Then

$$P(c_1|r) = \frac{P(r|c_1)\cdot P(c_1)}{P(r)} = \frac{p(1-p)^2 \times 0.1}{P(r)} = \frac{9}{640}\cdot\frac{1}{P(r)}$$

$$P(c_2|r) = \frac{P(r|c_2)\cdot P(c_2)}{P(r)} = \frac{(1-p)p^2 \times 0.9}{P(r)} = \frac{27}{640}\cdot\frac{1}{P(r)}$$

So, MED decodes r to c_2. But IMLD decodes r to c_1.

IMLD vs. MED

- IMLD maximizes $P(r|c)$. MED maximizes $P(c|r)$.
- (i) MED has the drawback that the decoding algorithm depends on the probability distribution of source messages.

  (ii) If all source messages are equally likely, then CMLD and MED are equivalent:

$$P(r|c_i) = P(c_i|r) \cdot P(c_i)/P(r) = P(c_i|r) \cdot \underbrace{\left[\frac{1}{M \cdot P(r)}\right]}_{\substack{\text{does not} \\ \text{depend on } c_i}}$$

  (iii) In practice IMLD (or CMLD) is used.
- In this course, we will use IMLD/CMLD.

1.3 Error Correcting & Detecting Capabilities of a Code

Detection Only

Strategy: If r is received, then accept r if and only if $r \in C$.

e-error detecting code

A code C is an **e-error detecting code** if the decoder always makes the correct decision if e or fewer errors per codeword are introduced by the channel.

Example:

Consider $C = \{000, 111\}$.

C is a 2-error detecting code.

C is not a 3-error detecting code.

Theorem 1.3

A code C of distance d is a $(d-1)$-error detecting code (but is not a d-error detecting code).

Proof:

Suppose $c \in C$ is sent.

If no errors occur, then c is received (and is accepted).

Suppose that # of errors is ≥ 1 and $\leq d-1$; let r be the received word. Then $1 \leq d(r,c) \leq d-1$, so $r \notin C$. Thus r is rejected. This proves that it is $(d-1)$-error detecting code.

Since $d(C) = d$, there exist $c_1, c_2 \in C$ with $d(c_1, c_2) = d$. If c_1 is sent and c_2 is received, then c_2 is accepted; the d errors go undetected. □

Correction

Strategy: IMLD/CMLD

e-error correcting code

A code C is an **e-error correcting code** if the decoder always makes the correct decision if e or fewer errors per codeword are introduced by the channel.

Example:

Consider $C = \{000, 111\}$.

C is a 1-error correcting code.

C is not a 2-error correcting code.

Theorem 1.4

A code C of distance d is an e-error correcting code, where $e = \lfloor \frac{d-1}{2} \rfloor$.

Proof:

Suppose that $c \in C$ is sent, at most $\frac{d-1}{2}$ errors are introduced, and r is received. Then $d(r,c) \leq \frac{d-1}{2}$.

On the other hand, if c_1 is any other codeword, then

$$\begin{aligned} d(r,c_1) &\geq d(c,c_1) - d(r,c) && \triangle \text{ ineq} \\ &\geq d - \frac{d-1}{2} && \text{since } d(C) = d \\ &= \frac{d+1}{2} \\ &> \frac{d-1}{2} \geq d(r,c) \end{aligned}$$

Hence c is the unique codeword at minimum distance from r, so the decoder correctly concludes that c was sent. □

Exercise:

Suppose $d(C) = d$, and let $e = \lfloor \frac{d-1}{2} \rfloor$. Show that C is *not* an $(e+1)$-error correcting code.

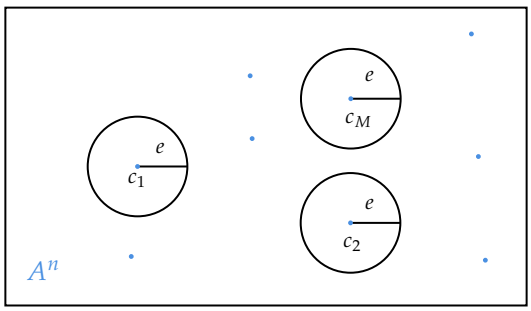
A natural question to ask is: given A, n, M, d, does there exist an $[n, M]$-code C over A of distance $\geq d$. This can be phrased as an equivalent sphere packing problem:

Sphere packing

Can we place M spheres of radius $e = \lfloor \frac{d-1}{2} \rfloor$ in A^n so that no two spheres overlap?

$C = \{c_1, \ldots, c_M\}$, $e = \lfloor \frac{d-1}{2} \rfloor$, S_c = sphere of radius e centered at c = all words within distance e of c.

We proved: if $c_1, c_2 \in C, c_1 \neq c_2$, then $S_{c_1} \cap S_{c_2} = \varnothing$.



Let $n = 128, q = 2, M = 2^{64}$. Does there exists a binary $[n, M]$-code with $d \geq 22$? If so, can encoding and decoding be done efficiently?

We'll view $\{0,1\}^{128}$ as a vector space of dimension 128 over $\mathbb{Z}_2$. We'll choose C to be a 64-dimensional subspace of this vector space. We will construct such a code at the end of the course. The main tools used will be linear algebra (over finite fields) and abstract algebra (rings and fields).

2

Introduction to Finite Fields

2.1 Definitions

ring

A **(commutative) ring** $(R, +, \cdot)$ consists of a set R and two operations $+ : R \times R \to R$ and $\cdot : R \times R \to R$, such that

1. $a + (b + c) = (a + b) + c \quad \forall a, b, c \in R.$
2. $a + b = b + a, \quad \forall a, b \in R.$
3. $\exists 0 \in R$ such that $a + 0 = a, \forall a \in R.$
4. $\forall a \in R, \exists -a \in R$ such that $a + (-a) = 0.$
5. $a \cdot (b \cdot c) = (a \cdot b) \cdot c, \quad \forall a, b, c \in R.$
6. $a \cdot b = b \cdot a, \qquad \forall a, b \in R.$
7. $\exists 1 \in R, 1 \neq 0,$ such that $a \cdot 1 = a, \quad \forall a \in R.$
8. $a \cdot (b + c) = a \cdot b + b \cdot c, \qquad \forall a, b, c \in R.$

Notation We will denote $(R, +, \cdot)$ by R.

Example:
$\mathbb{Q}, \mathbb{R}, \mathbb{C}, \mathbb{Z}$ are commutative rings.

field

A **field** $(F, +, \cdot)$ is a commutative ring with the additional property:

9. $\forall a \in F, a \neq 0, \exists a^{-1} \in F$ such that $a \cdot a^{-1} = 1.$

Example:
$\mathbb{Q}, \mathbb{R}, \mathbb{C}$ are fields. $\mathbb{Z}$ is *not* a field.

infinite/finite field

A field $(F, +, \cdot)$ is a **finite field** if F is a finite set; otherwise it is an **infinite field**. If F is a finite field , its **order** is $|F|$.

Example:

$\mathbb{Q}, \mathbb{R}, \mathbb{C}$ are infinite fields.

For which integers $n \geq 2$ does there *exist* a finite field of order n? How does one *construct* such a field, i.e., what are the field elements, and how are the field operations performed?

The Integers Modulo n

Let $n \geq 2$. Recall that $\mathbb{Z}_n$ consists of the set of equivalence classes of integers modulo n, $\mathbb{Z}_n = \{[0], [1], [2], \ldots, [n-1]\}$, with addition and multiplication: $[a] + [b] = [a+b]$, $[a] \cdot [b] = [a \cdot b]$.

More simply, we write $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$, and perform addition and multiplication modulo n.

Example:

$\mathbb{Z}_9 = \{0, 1, 2, \ldots, 8\}$. In $\mathbb{Z}_9$, $3 + 7 = 1$ and $3 \cdot 7 = 3$.

More precisely, $3 + 7 \equiv 1 \pmod 9$ and $3 \cdot 7 \equiv 3 \pmod 9$.

$\mathbb{Z}_n$ is a commutative ring (i.e., axioms 1-8 in the definition are satisfied).

When is $\mathbb{Z}_n$ is a field?

Theorem 2.1

$\mathbb{Z}_n$ is a field if and only if n is prime.

Proof:

$\Leftarrow$ Suppose n is prime. Let $a \in \mathbb{Z}_n$, $a \neq 0$ (so $1 \leq a \leq n-1$). Since n is prime, $\gcd(a, n) = 1$. Hence $\exists s, t \in \mathbb{Z}$ such that $as + nt = 1$. Reducing both sides modulo n gives $as \equiv 1 \pmod n$. Hence $a^{-1} = s$. Thus $\mathbb{Z}_n$ is a field.

$\Rightarrow$ Suppose n is composite, say $n = ab$ where $2 \leq a, b \leq n-1$.

Now, if a^{-1} exists, say $ac \equiv 1 \pmod n$, then $abc \equiv b \pmod n$, so $nc \equiv b \pmod n$. Thus $b \equiv 0 \pmod n$, so $n \mid b$ which is absurd since $2 \leq b \leq n-1$. Thus $\mathbb{Z}_n$ is not a field.

□

We have established the existence of finite fields of order n, for each prime n. What about finite fields of order n, where n is composite? In particular, is there a field of order 4? Order 6?

characteristic

Let F be a field. The **characteristic** of F, denoted $\text{char}(F)$, is the smallest positive integer m such that $\underbrace{1 + \cdots + 1}_{m} = 0$. If no such m exists, then $\text{char}(F) = 0$.

Example:

$\mathbb{Q}, \mathbb{R}, \mathbb{C}$ have characteristic 0. $\mathbb{Z}_p$ (p prime) has characteristic p.

Theorem 2.2

If $\text{char}(F) = 0$, then F is an infinite field.

Proof:
The elements $1, 1+1, 1+1+1, \ldots$ are distinct, because if $\underbrace{1+\cdots+1}_{a} = \underbrace{1+\cdots+1}_{b}$ where $a < b$, then $(\underbrace{1+\cdots+1}_{b}) - (\underbrace{1+\cdots+1}_{a}) = 1+\cdots+1_{b-a} = 0$, contradicting $\text{char}(F) = 0$. □

Theorem 2.3

Let F be a field with $\text{char}(F) = m \neq 0$. Then m is prime.

Proof:
Suppose m is composite, say $m = ab$ where $2 \leq a, b \leq m-1$. Let $s = \underbrace{1+\cdots+1}_{a}$ and $t = \underbrace{1+\cdots+1}_{b}$; note that $s, t \neq 0$. Then $s \cdot t = (\underbrace{1+\cdots+1}_{a}) \cdot (\underbrace{1+\cdots+1}_{b}) = \underbrace{1+\cdots+1}_{ab=m} = 0$. Thus

$$s \cdot t \cdot t^{-1} = s \cdot 1 = s = 0,$$

a contradiction. We then conclude that m is prime. □

Let F be a finite field of characteristic p. Consider the subset of elements of F:

$$E = \{0, 1, 1+1, 1+1, \ldots, \underbrace{1+\cdots+1}_{p-1}\}.$$

The elements of E are distinct. One can verify that E is a field, using the same operations as F. E is a **subfield** of F. If we identify the elements of E with the elements of $\mathbb{Z}_p$ in the natural way, then E is essentially the same field as $\mathbb{Z}_p$. We have proven:

Theorem 2.4

Let F be a finite field of char p. Then $\mathbb{Z}_p$ is a subfield of F.

Finite fields as vector spaces

Let F be a finite field of characteristic p. Identify:

vectors	$\leftrightarrow$	elements of F
scalars	$\leftrightarrow$	elements of $\mathbb{Z}_p$
vector addition	$\leftrightarrow$	addition of F
scalar multiplication	$\leftrightarrow$	multiplication of F

Then F is a vector space over $\mathbb{Z}_p$ (i.e., the axioms of what it means to be a vector space are satisfied).

2.2 Finite fields: Non-existence

Theorem 2.5

Let F be a finite field of characteristic p. Then the order of F is p^n, for some positive integer n.

Proof:
Let the dimension of F as a vector space over $\mathbb{Z}_p$ be n. Let $\alpha_1, \ldots, \alpha_n$ be a basis for F over $\mathbb{Z}_p$. Then each element $\beta \in F$ can be written uniquely in the form $\beta = c_1\alpha_1 + \cdots + c_n\alpha_n$, where $c_i \in \mathbb{Z}_p$. Thus $F = \left\{ \sum_{i=1}^{n} c_i\alpha_i : c_i \in \mathbb{Z}_p \right\}$, so $|F| = p^n$. □

For example, there do not exist finite fields of order $6, 10, 12, 14, 15, \ldots$

Do finite fields of orders $4, 8, 9, 16, 25, 27, \ldots$ exist?

2.3 Existence of finite fields

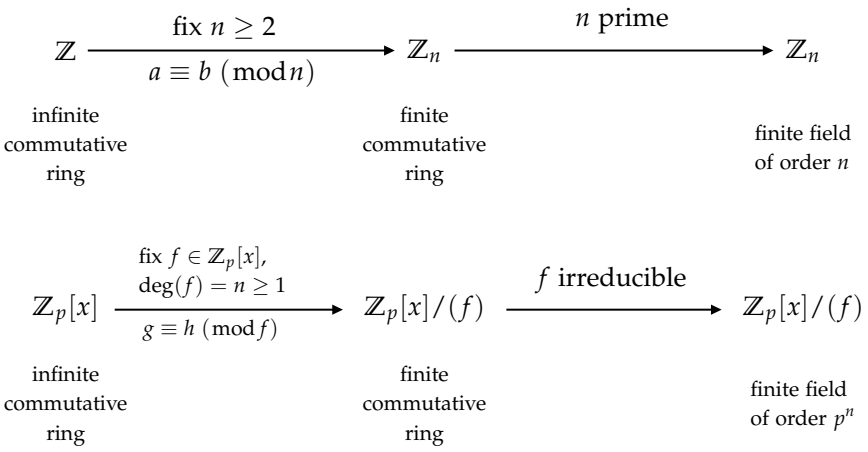
Polynomial rings Let F be a field. $F[x]$ denotes the set of all polynomials in x with coefficients from F. Addition and multiplication of polynomials in $F[x]$ is done in the usual way, with coefficient arithmetic done in F.

Example:
In $\mathbb{Z}_5[x]$,

$$(3x^4 + 2x^3 + x + 4) + (x^5 + 2x^4 + x^2 + 2x + 3) = x^5 + 2x^3 + x^2 + 3x + 2$$
$$(3x^2 + 4x + 1) \cdot (2x^2 + x + 2) = (x^4 + x^3 + 2x^2 + 4x + 2).$$

Note that $F[x]$ is an infinite commutative ring.

Construction of finite fields: main idea



Polynomial division

Let $f, g \in F[x]$, with $g \neq 0$. Then there exist unique polynomials $r, s \in F[x]$ such that $f = sg + r$, $\deg(r) < \deg(g)$. By convention, $\deg(0) = -\infty$.

(s: quotient; r: remainder)

Example:

Consider $f = 3x^4 + 2x^2 + x + 1, g = 2x^2 + 3x + 4 \in \mathbb{Z}_5[x]$.

$$\begin{array}{rl|lllll} & & 4x^2 & +3 & & & \\ \hline 2x^2+3x+4 &) & 3x^4 & +2x^3 & +2x^2 & +x & +1 \\ & & 3x^4 & +2x^3 & +x^2 & & \\ \hline & & & & x^2 & +x & +1 \\ & & & & x^2 & +4x & +2 \\ \hline & & & & & 2x & +4 \end{array}$$

So, $f = (4x^2 + 3)g + (2x + 4)$.

The ring $F[x]/(f)$

$g \equiv h \pmod f$

Let $f \in F[x]$ with $\deg(f) \geq 1$. Let $g, h \in F[x]$. Then g is congruent to h modulo f, written $g \equiv h \pmod f$, if $g - h = \ell f$ for some $\ell \in F[x]$ (equivalently, $f \mid (g - h)$, or g, h leave the same remainder upon division by f).

The relation $\equiv \pmod f$ is an equivalence relation, and partitions $F[x]$ into equivalence classes:

$$[g] = \{h \in F[x] : g \equiv g \pmod f\}.$$

Addition & multiplication: $[g] + [h] = [g + h], [g] \cdot [h] = [g \cdot h]$.

$F[x]/(f)$

The set of equivalence classes is denoted $F[x]/(f)$.

Theorem 2.6

$F[x]/(f)$ is a commutative ring.

Suppose now that $\deg(f) = n$. Let $g \in F[x]$. Then we can write $g = sf + r$, where $s, r \in F[x]$, and $\deg(r) < n$. Thus $g \equiv r \pmod f$, so $[g] = [r]$.

If $r_1, r_2 \in F[x], r_1 \neq r_2$, $\deg(r_1), \deg(r_2) < n$, then $f \nmid (r_1 - r_2)$, so $r_1 \not\equiv r_2 \pmod f$. Thus $[r_1] \neq [r_2]$.

Thus the polynomials in $F[x]$ of degree $< n$ are a complete set of representatives of the equivalence classes of $F[x]/(f)$.

Now, let $F = \mathbb{Z}_p$. Then $\mathbb{Z}_p[x]/(f) = \{[r] : r \in \mathbb{Z}_p[x], \deg(r) < n\}$. Thus $|\mathbb{Z}_p[x]/(f)| = p^n$, so $\mathbb{Z}_p[x]/(f)$ is a commutative ring of order p^n.

When is $F[x]/(f)$ a field?

irreducible over F

Let $f \in F[x]$, with $\deg(f) \geq 1$. Then **irreducible over** F if f cannot be written as $f = g \cdot h$, $g, h \in F[x]$, $\deg(g) \geq 1, \deg(h) \geq 1$.

Example:

x^2+1 is irreducible over $\mathbb{R}$, since it has no roots in $\mathbb{R}$.

x^2+1 is reducible over $\mathbb{C}$, since $x^2+1=(x+i)(x-i)$.

x^2+1 is reducible over $\mathbb{Z}_2$, since $x^2+1=(x+1)(x+1)$.

x^2+1 is irreducible over $\mathbb{Z}_3$, since it has no roots in $\mathbb{Z}_3$.

Theorem 2.7

$F[x]/(f)$ is a field if and only if f is irreducible over F.

Proof:
Analogous to the proof of the theorem: $\mathbb{Z}_n$ is a field if and only if n is prime. □

Now let's construct finite fields.

Theorem 2.8

Let $f \in \mathbb{Z}_p[x]$ be an irreducible polynomial of degree $n \geq 1$. Then $\mathbb{Z}_p[x]/(f)$ is a finite field of order p^n and characteristic p. The elements are the polynomials in $\mathbb{Z}_p[x]$ of degree $< n$.

Example: finite field of order $4 = 2^2$

Here $p=2$ and $n=2$. Let $f(x)=x^2+x+1 \in \mathbb{Z}_2[x]$. Then $f(0)=1$, $f(1)=1$, so f has no roots in $\mathbb{Z}_2$. Thus f is irreducible over $\mathbb{Z}_2$.

So, $F=\mathbb{Z}_2[x]/(x^2+x+1)$ is a finite field of order $2^2=4$. The elements are $\{0,1,x,x+1\}$ where $[\,]$ is omitted.

Example of addition: $x+(x+1)=1$

Example of multiplication: $x\cdot(x+1)=x^2+x=1$

Example: finite field of order $2^3 = 8$

Here, $p=2$ and $n=3$. We need an irreducible polynomial. We need an irreducible polynomial of degree 3 over $\mathbb{Z}_2$.

Candidates: $x^3, x^3+1, x^3+x, x^3+x+1, x^3+x^2, x^3+x^2+1, x^3+x^2+x, x^3+x^2+x+1$

- Try $f(x)=x^3+x+1$.

  Since $f(0)=f(1)=1$, f has no roots in $\mathbb{Z}_2$, and thus no linear factors in $\mathbb{Z}_2[x]$. Thus f is irreducible over $\mathbb{Z}_2$, and $F_1=\mathbb{Z}_2[x]/(x^3+x+1)$ is a finite field of order $2^3=8$.

  The elements of F_1 are $\{0,1,x,x+1,x^2,x^2+1,x^2+x,x^2+x+1\}$.

  Example of addition: $(x^2+x)+(x^2+x+1)=1$.

  Example of multiplication: $(x^2+x)\cdot(x^2+x+1)=x^4+x=x^2$.

  Example of inversion: $x^{-1}=x^2+1$, since $x\cdot(x^2+1)=1$.

- x^3+x^2+1 is irreducible over $\mathbb{Z}_2$, so $F_2=\mathbb{Z}_2[x]/(x^3+x^2+1)$ is a finite field of order 8. The elements of F_2 are $\{0,1,x,x+1,x^2,x^2+1,x^2+x,x^2+x+1\}$.

Note that F_1 and F_2 are not the same field. For example, in F_1, $x\cdot x^2=x+1$, whereas in F_2, $x\cdot x^2=x^2+1$. However, F_1 and F_2 are isomorphic (essentially the same). Formally, there is a bijection $\phi: F_1 \to F_2$ such that $\phi(a+b)=\phi(a)+\phi(b)$ and $\phi(a\cdot b)=\phi(a)\cdot\phi(b)$ $\forall a,b \in F_1$.

Existence and uniqueness of finite fields

Let p be prime and $n \geq 1$. Then there exists an irreducible polynomial of degree n over $\mathbb{Z}_p$.

Theorem 2.9

There exists a finite field of order q if and only if $q = p^n$ for some prime p and $n \geq 1$.

Actually, any two finite fields of the same order are isomorphic.

We will denote *the* finite field of order q by $\mathrm{GF}(q)$ "the Galois Field of order q".

In the previous example, we saw two ways of representing the finite field $\mathrm{GF}(2^3)$.

2.4 Properties of finite fields

Theorem 2.10: Frosh's dream

Let F be a finite field of characteristic p, and let $\alpha, \beta \in F$. Then $(\alpha+\beta)^{p^m} = \alpha^{p^m} + \beta^{p^m}$ $\forall m \geq 1$.

Proof ($m = 1$):
By the Binomial Theorem,

$$(\alpha+\beta)^p = \binom{p}{0}\alpha^p + \sum_{i=1}^{p-1}\binom{p}{i}\alpha^i\beta^{p-i} + \binom{p}{p}\beta^p.$$

Now for $1 \leq i \leq p-1$,

$$\binom{p}{i} = \frac{p(p-1)(p-2)\cdots(p-i+1)}{1\cdot 2\cdot 3\cdots i} \equiv 0 \pmod{p},$$

since p divides the numerator but not the denominator, and since $\binom{p}{i}$ is an integer. Thus

$$\binom{p}{i}\alpha^i\beta^{p-i} = \underbrace{\alpha^i\beta^{p-i} + \cdots + \alpha^i\beta^{p-i}}_{\binom{p}{i}} = (1+\cdots+1)\alpha^i\beta^{p-i} = 0.$$

Hence $(\alpha+\beta)^p = \alpha^p + \beta^p$. The statement for $m \geq 1$ can be proven by induction. □

The multiplicative group $\mathrm{GF}(q)^*$

multiplicative group of $\mathrm{GF}(q)$

The **multiplicative group of** $\mathrm{GF}(q)$ is $\mathrm{GF}(q)^* = \mathrm{GF}(q) \setminus \{0\}$.

Theorem 2.11

Let $\alpha \in \mathrm{GF}(q)^*$. Then $\alpha^{q-1} = 1$.

Note that if $\mathrm{GF}(q) = \mathbb{Z}_p$, this is Fermat's Little Theorem.

Proof:
Let the (distinct) elements of $\mathrm{GF}(q)^*$ be $\alpha_1, \alpha_2, \ldots, \alpha_{q-1}$. Consider the (nonzero) elements $\alpha\alpha_1, \alpha\alpha_2, \ldots, \alpha\alpha_{q-1}$. These elements are distinct because if $\alpha\alpha_i = \alpha\alpha_j$ for some $i \neq j$, then $\alpha^{-1}(\alpha\alpha_i) = \alpha^{-1}(\alpha\alpha_j)$, so $\alpha_i = \alpha_j$, a contradiction. Hence $\{\alpha\alpha_1, \alpha\alpha_2, \ldots, \alpha\alpha_{q-1}\} = \{\alpha_1, \alpha_2, \ldots, \alpha_{q-1}\}$, so

$$(\alpha\alpha_1)(\alpha\alpha_2)\cdots(\alpha\alpha_{q-1}) = \alpha_1\alpha_2\cdots\alpha_{q-1}.$$

Cancelling gives $\alpha^{q-1} = 1$. □

Corollary 2.12

Let $\alpha \in \mathrm{GF}(q)$. Then $\alpha^q = \alpha$.

Order of finite elements

order of α

Let $\alpha \in \mathrm{GF}(q)^*$. The **order of** α, denoted $\mathrm{ord}(\alpha)$, is the smallest positive integer t such that $\alpha^t = 1$.

Theorem 2.13

Let $\alpha \in \mathrm{GF}(q)^*$, $\mathrm{ord}(\alpha) = t$. Then $\alpha^s = 1$ if and only if $t \mid s$.

Proof:
Let $s \in \mathbb{Z}$. Then long division of s by t yields

$$s = \ell t + r, \quad \text{where } 0 \leq r < t.$$

Now, $\alpha^s = \alpha^{\ell t + r} = (\alpha^t)^\ell \cdot \alpha^r = \alpha^r$. Hence $\alpha^s = 1 \iff \alpha^r = 1 \iff r = 0 \iff t \mid s$. □

Corollary 2.14

Let $\alpha \in \mathrm{GF}(q)^*$. Then $\mathrm{ord}(\alpha) \mid (q-1)$.

Example:
There is only one element in $\mathrm{GF}(q)$ of order 1, namely the element 1.

Example:
Consider $\mathrm{GF}(2^3) = \mathbb{Z}_2[x]/(x^3 + x + 1)$. The order of $\alpha = x^2 + 1$ is 7.

Example:
Consider $\mathrm{GF}(2^4) = \mathbb{Z}_2[x]/(x^4 + x + 1)$.

$f(x) = x^4 + x + 1$ has no roots in $\mathbb{Z}_2$, thus no linear factors. Also, $f(x)$ has no irreducible quadratic factors, since $(x^2 + x + 1) \nmid f(x)$. Note here $x^2, x^2 + 1, x^2 + x$ are reducible quadratic polynomials. Thus f is irreducible over $\mathbb{Z}_2$.

Find $\mathrm{ord}(x)$ in $\mathrm{GF}(2^4)$.

Solution: We have $x^1 = x, x^2 = x^2, x^3 = x^3, x^4 = x + 1, x^5 = x^2 + x \neq 1$. Thus $\mathrm{ord}(x) \neq 1, 3, 5$. Since $\mathrm{ord}(x) \mid 15$, we must have $\mathrm{ord}(x) = 15$.

Let $\alpha \in \mathrm{GF}(q)^*$ with $\mathrm{ord}(\alpha) = t$. Then the elements $\alpha^0, \alpha^1, \alpha^2, \ldots, \alpha^{t-1}$ are distinct. In particular, if

$\text{ord}(\alpha) = q - 1$, then $\text{GF}(q)^* = \{\alpha^0, \alpha^1, \alpha^2, \ldots, \alpha^{q-2}\}$.

generator

A **generator** of $\text{GF}(q)^*$ is an element of order $q - 1$.

Example:

$\alpha = x$ is a generator of $\text{GF}(2^4) = \mathbb{Z}_2[x]/(x^4 + x + 1)$ since $\text{ord}(x) = 15$. Let's verify the above fact:

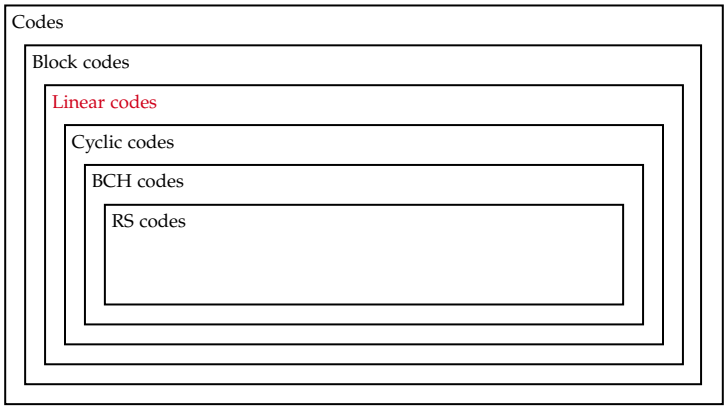
$$
\begin{array}{llll}
x^0 = 1, & x^1 = x, & x^2 = x^2, & x^3 = x^3, \\
x^4 = x + 1, & x^5 = x^2 + x, & x^6 = x^3 + x^2, & x^7 = x^3 + x + 1, \\
x^8 = x^2 + 1, & x^9 = x^3 + x, & x^{10} = x^2 + x + 1, & x^{11} = x^3 + x^2 + x, \\
x^{12} = x^3 + x^2 + x + 1, & x^{13} = x^3 + x^2 + x + 1, & x^{14} = x^3 + 1, & x^{15} = 1.
\end{array}
$$

Theorem 2.15

Every finite field $\text{GF}(q)$ has a generator.

3

Linear codes



3.1 Definition

Let $F = \mathrm{GF}(q)$. Let $V_n(F) = \underbrace{F \times \cdots \times F}_{n}$. $V_n(F)$ is an n-dimensional space over F. $|V_n(F)| = q^n$.

linear (n,k)-code over F

A **linear (n,k)-code over** F is a k-dimensional subspace of $V_n(F)$.

Recall a *subspace* S of a vector space V over F is non-empty subset $S \subseteq V$ such that:

$$\text{(i) } a, b \in S \implies a + b \in S, \text{ (ii) } a \in S, \lambda \in F \implies \lambda a \in S.$$

If S is a subspace of V, then S is itself a vector space over F; also $0 \in S$. A *basis* of S is a linearly independent, spanning subset of S. All bases of S have the same cardinality, called the *dimension* of S.

3.2 Properties of Linear Codes

Let C be an (n,k)-code over F, and let $v_1, v_2, \ldots, v_k$ be an ordered basis for C.

1. Number of codewords

The elements of C are precisely

$$c_1v_1 + c_2v_2 + \cdots + c_kv_k, \quad c_i \in F.$$

Thus, $|C| = M = q^k$.

2. Rate

The rate of C is $R = \frac{\log_q M}{n} = \frac{\log_q q^k}{n} = \frac{k}{n}$.

3. Weight

Hamming weight

The **Hamming weight** $w(v)$ of a vector $v \in V_n(F)$ is the number of nonzero coordinates in v. The **Hamming weight** of a linear code C is $w(C) = \min\{w(c) : c \in C, c \neq 0\}$.

Theorem 3.1

If C is a linear code, then $w(c) = d(C)$.

Proof:
We have

$$\begin{aligned} d(C) &= \min\{d(x,y) : x,y \in C, x \neq y\} \\ &= \min\{w(x-y) : x,y \in C, x \neq y\} && \text{since } d(x,y) = w(x-y) \\ &= \min\{w(c) : c \in C, c \neq 0\} && \text{since } C \text{ is linear }, x-y \in C \\ &= w(C). \end{aligned}$$

4. Encoding

Since there are q^k codewords, there are also q^k source messages. We shall assume that source messages are the elements of F^k. Then a convenient and natural bijection (i.e., *encoding rule*) between F^k and C is defined by:

$$\boxed{m = (m_1, m_2, \ldots, m_k) \mapsto c = m_1v_1 + m_2v_2 + \cdots + m_kv_k.}$$

Note that different ordered bases for C yield different encoding rules.

5. Generator matrix

A convenient way to represent C.

generator matrix

A **generator matrix** G for an (n,k)-code C is a $k \times n$ matrix whose rows form a basis for C:

$$G = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{bmatrix}_{k \times n.}$$

Note that the encoding rule is $c = mG$.

Example: linear code

Consider the $(5,3)$-binary code:

$$C = \langle \underset{c_1}{10011}, \underset{c_2}{01001}, \underset{c_3}{00110} \rangle$$

and c_1, c_2, c_3 are linearly independent over GF(2).

A generator matrix for C is $G = \left[\begin{array}{ccc|cc} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{array}\right]_{3\times 5}$

The encoding rule (with respect to the ordered basis $\{c_1, c_2, c_3\}$) is $c = mG$.

000	$\to$	00000	100	$\to$	10011
001	$\to$	00110	101	$\to$	10101
010	$\to$	01001	110	$\to$	11010
011	$\to$	01111	111	$\to$	11100

Other properties: $M = |C| = 2^3 = 8$, $R = 3/5$, $d(C) = w(C) = 2$.

Standard form GM

standard form generator matrix

Let C be an (n,k)-code over F. A GM G for C of the form $G = \left[I_k | A\right]_{k\times n}$ is said to be in **standard form**.

systematic code

If C has a GM in standard form, then C is a **systematic code**.

Example: systematic/non-systematic code

$C = \langle 100011, 001001, 000110 \rangle$ is a *non-systematic* $(6,3)$-binary code.

But $C' = \langle 10011, 001001, 010010 \rangle$ is *systematic*. A GM for C' is

$$G = \left[\begin{array}{cccccc} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

equivalent codes

Two codes C, C' over F are **equivalent** if C' can be obtained from C by choosing a permutation of the coordinate positions $\{1, 2, \ldots, n\}$, and then consistently rearranging every codeword of C according to this permutation.

Below are some facts of equivalent codes:

1. If C is linear, and C' is equivalent to C, then C' is linear.
2. Equivalent codes have the same length, dimension, distance.
3. Every linear code is equivalent to a systematic code.

3.3 The Dual Code

inner product

Let $x = (x_1, x_2, \ldots, x_n), y = (y_1, y_2, \ldots, y_n) \in V_n(F)$. The **inner product** of x and y is

$$x \cdot y = \sum_{i=1}^{n} x_i y_i \in F$$

For all $x, y, z \in V_n(F)$ and $\lambda \in F$:

1. $x \cdot y = y \cdot x$.
2. $x \cdot (y + z) = x \cdot y + x \cdot z$.
3. $(\lambda x) \cdot y = \lambda(x \cdot y)$
4. $x \cdot x$ does *not* imply that $x = 0$.

Example:

Consider $x = 111100 \in V_6(\mathbb{Z}_2)$. Then $x \cdot x = 0$, but $x \neq 0$. More generally, if $x \in V_n(\mathbb{Z}_2)$, then $x \cdot x = 0$ if and only if $w(x)$ is even.

orthogonal vectors

Two vectors x, y are **orthogonal** if $x \cdot y = 0$.

dual code

Let C be an (n, k)-code over F. The **dual code** or **orthogonal code** of C is

$$C^{\perp} = \{x \in V_n(F) : x \cdot y = 0, \quad \forall y \in C\}.$$

Theorem 3.2

If C is an (n, k)-code over F, then $C^{\perp}$ is an $(n, n-k)$-code over F.

Proof:

Let G be a GM for C, and let the rows of G be $v_1, v_2, \ldots, v_k$.

Claim Let $x \in V_n(F)$. Then $x \in C^{\perp}$ if and only if $v_1 \cdot x = v_2 \cdot x = \cdots = v_k \cdot x = 0$.

Let's prove the claim.

($\Rightarrow$) is clear since $v_1, v_2, \ldots, v_k \in C$.

($\Leftarrow$) Suppose $v \in C$. Then we can write $v = \lambda_1 v_1 + \cdots + \lambda_k v_k$, where $\lambda_i \in F$. Then

$$v \cdots x = (\lambda_1 v_1 + \cdots + \lambda_k v_k) \cdot x = \lambda_1(v_1 \cdot x) + \cdots + \lambda_k(v_k \cdot x) = 0.$$

Thus, $C^{\perp} = \{x \in V_n(F) : Gx^T = 0\}$ = null space of G. Since G has rank k, $C^{\perp}$ is a subspace of $V_n(F)$ of dimension $n - k$. □

3.4 Parity-Check Matrix

Let G be a GM for a linear code C. Then $C^\perp$ = null space of G.

Theorem 3.3

If C is a linear code, then $(C^\perp)^\perp = C$.

Proof:
Let C be an (n,k)-code. Then $C^\perp$ is an $(n, n-k)$-code.

Furthermore, $(C^\perp)^\perp$ is an (n,k)-code, and $C \subseteq (C^\perp)^\perp$.

Since $\dim(C) = \dim\big((C^\perp)^\perp\big)$, it follows that $C = (C^\perp)^\perp$. □

parity-check matrix

If C is linear, then a generator matrix H for $C^\perp$ is called a **parity-check matrix** (PCM) for C.

Note:
H is an $(n-k) \times n$ matrix.

C has many PCMs.

Constructing a GM for $C^\perp$

Theorem 3.4

Let C be an (n,k)-code with GM $G = [I_k \mid A]$. Then $H = [-A^T \mid I_{n-k}]$ is a GM for $C^\perp$.

Note that A is $k \times (n-k)$ matrix.

Proof:
Since $\text{rank}(H) = n-k$, H is a GM for an $(n, n-k)$-code $\overline{C}$. Also,

$$GH^T = [I_k|A]\begin{bmatrix} -A \\ \hline I_{n-k} \end{bmatrix} = -A + A = 0.$$

Thus $\overline{C} \subseteq C^\perp$. Since $\dim(\overline{C}) = \dim(C^\perp)$, we have $\overline{C} = C^\perp$. Hence H is a GM for $C^\perp$. □

Example:

Consider the $(5,2)$-code C over $\mathbb{Z}_3$ with GM $G = \begin{bmatrix} 2 & 0 & 2 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}$. Find a PCM for C.

Solution Find a GM for C in standard form:

$$G \xrightarrow{R_1 \leftarrow 2R_1} \begin{bmatrix} 1 & 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \xrightarrow{R_2 \leftarrow R_2 - R_1} \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 2 & 0 \\ 0 & 1 & 2 & 1 & 1 \end{array}\right].$$

So,

$$H = \left[\begin{array}{cc|ccc} 2 & 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{array}\right]$$

is a PCM for C.

We have

$$C = \left\{ \begin{array}{lll} 00000, & 20210, & 10120, \\ 11001, & 22002, & 01211, \\ 02122, & 21121, & 12212 \end{array} \right\}$$

Thus $d(C) = w(C) = 3, R = 2/5$.

Notes on PCMs

Let C be an (n,k)-code over F with GM G.

1. An $(n-k) \times n$ matrix H over F is a PCM for C iff $GH^T = 0$ and $\text{rank}(H) = n-k$.
2. G is a PCM for $C^\perp$ (since $(C^\perp)^\perp = C$).
3. $C = \text{null}(H)$.
4. Let H be a PCM for C, and let $x \in V_n(F)$, then $x \in C$ iff $Hx^T = 0$.

3.5 Distance of a Linear Code

Theorem 3.5: distance of a linear code

Let H be a PCM for a (n,k)-code C over F. Then $d(C) \geq s$ if and only if every $s-1$ columns of H are linearly independent over F.

Corollary 3.6

Let H be a PCM for a linear code C over F. Then $d(C)$ is the smallest number of columns of H that are *linearly dependent* over F.

3.6 Hamming Codes

Hamming code of order r over $F = \text{GF}(q)$

A **Hamming code of order r over $F = \text{GF}(q)$** is an (n,k)-coder over F with $n = \frac{q^r-1}{q-1}$ and $k = n-r$, and with PCM H_r and $r \times n$ matrix whose columns are nonzero, and no two of whose columns are scalar multiples of each other.

Notes on Hamming codes:

1. If $v \in V_r(F)$, $v \neq 0$, then *exactly one* scalar multiple of v is a column of H (giving $n = \frac{q^r-1}{q-1}$ columns in total).
2. H_r has rank r, since among its columns are scalar multiples of the unit vectors. Hence a Hamming code of order r over $\text{GF}(q)$ does indeed have dimension $k = n-r$.
3. A Hamming code of order r over $\text{GF}(q)$ has distance 3 (by design), so is a single-error correcting code.

3.7 Decoding Single-Error Correcting Codes

Let H be a PCM for an (n, k, d)-code C over F with $d \geq 3$.

error vector

Suppose $c \in C$ is sent, and $r \in V_n(F)$ is received, The **error vector** is $e = r - c$ (so $r = c + e$).

Algorithm 1: Decoding algorithm for single-error correcting codes

Input: PCM H and a received word r

Compute $s = Hr^T$.
if $s = 0$ **then** accept r as transmitted code $(e = 0)$
if $s \neq 0$ **then**
    Compare s with the columns of H.
    if $s = \alpha h_i$ *for some* i **then**
        Set $e = (0, \ldots, 0, \alpha, 0, \ldots, 0)$ where α is at i^{th} position.
        Decode r to $c = r - e$.
else
    Report that more than one error has occurred.

Note that if $w(e) = 0$ or $w(e) = 1$, then the decoding algorithm is *guaranteed* to make the correct decision.

3.8 Perfect Codes

perfect code

Let C be an $[n, M]$-code of distance d over A, with $|A| = q$ and $e = \lfloor \frac{d-1}{2} \rfloor$. Then C is **perfect** if each $x \in A^n$ is in the sphere of radius e centered at some $c \in C$.

Equivalently, C is perfect if

$$M \cdot \sum_{i=0}^{e} \binom{n}{i} (q-1)^i = q^n \, .$$

For fixed q, n, d, a perfect code has maximum possible M. In other words, a perfect code has maximum possible rate $R = \frac{\log_q M}{n}$, for fixed q, n, d.

Some facts:

- $C = A^n$ is a (trivial) perfect code with distance $d = 1$.
- Let n be odd. Binary replication code is a perfect code with $d = n$.
- Every perfect code has odd distance.
- For a perfect code, IMLD = CMLD.
- All Hamming codes of order r over GF(q) are perfect.

Theorem 3.7: Tietäräinen, 1973

The only perfect codes are

1) $V_n(\mathrm{GF}(q))$
2) The binary replication code of odd length.
3) The $(23, 12, 7)$-binary Golay code and all codes equivalent to it.
4) The $(11, 6, 5)$-ternary[a] Golay and all codes equivalent to it.

   A GM is

$$G = \left[\begin{array}{c|ccccc} & 1 & 1 & 1 & 1 & 1 \\ & 0 & 1 & 2 & 2 & 1 \\ & 1 & 0 & 1 & 2 & 2 \\ I_6 & 2 & 1 & 0 & 1 & 2 \\ & 2 & 2 & 1 & 0 & 1 \\ & 1 & 2 & 2 & 1 & 0 \end{array}\right]_{6\times 11}$$

5) The Hamming codes and all codes of the same $[n, M, d]$ parameters as them. ($d = 3$).

[a] over $\mathbb{Z}^3$

3.9 Syndrome Decoding

Let C be an (n, k)-code over $F = \mathrm{GF}(q)$ with PCM H.

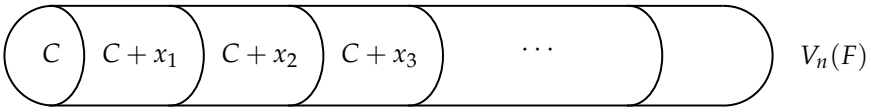
$x \equiv y \pmod{C}$

Let $x, y \in V_n(F)$. We write $x \equiv y \pmod{C}$ if $x - y \in C$.

Facts:

1. $\equiv \pmod{C}$ is an *equivalence relation*.
2. The set of equivalences classes partitions $V_n(F)$.
3. The equivalence class containing $x \in V_n(F)$ is called a **coset** of C. Check the broader definition in PMATH 347. This class is

$$C + x = \{y \in V_n(F) : y \equiv \pmod{C}\} = \{c + x : c \in C\}.$$



Example: Cosets

Consider a $(5, 2)$-binary code C with GM

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

Find all cosets of C.

Solution The cosets of C are:

$$\begin{aligned}
C = C + 00000 &= \{00000, 10111, 01110, 11001\} = C + 10111 = C + 01110 + C + 11001 \\
C + 10000 &= \{10000, 00111, 11110, 01001\} = C + 00111 = C + 11110 = C + 01001 \\
C + 01000 &= \{01000, 11111, 00110, 10001\} \\
C + 00100 &= \{00100, 10011, 01010, 11101\} \\
C + 00010 &= \{00010, 10101, 01100, 11011\} \\
C + 00001 &= \{00001, 10110, 01111, 11000\} \\
C + 10100 &= \{10100, 00011, 11010, 01101\} \\
C + 10010 &= \{10010, 00101, 11100, 01011\}
\end{aligned}$$

Facts on cosets:

1. $C + 0 = C$.
2. If $y \in C + x$, then $C + y = C + x$.
3. All cosets of C has the same size q^k.
4. The number of cosets is $q^n / q^k = q^{n-k}$.

syndrome

Let H be a PCM for an (n, k)-code C over F. For $x \in V_n(F)$, the **syndrome** of x (with respect to H) is $s = Hx^T$.

Theorem 3.8

Let $x, y \in V_n(F)$. Then $x \equiv y \pmod{C}$ if $Hx^T = Hy^T$.

Syndrome decoding algorithm

For each coset of C, select an *arbitrary* vector of smallest weight in that coset, and call it *coset leader* of that coset. Store a table of coset leaders and their syndromes.

Algorithm 2: Decoding algorithm (CMLD)

1 Given r, compute $s = Hr^T$.
2 Let e be the corresponding coset leader.
3 Decode r to $c = r - e$.

The decoding algorithm is guaranteed to make the correct decision if the error vector is a coset leader; otherwise is guaranteed to make a wrong decision.

Theorem 3.9

Let C be an (n, k)-code over F with distance d. Let $x \in V_n(F)$ be a vector of weight $\leq \lfloor \frac{d-1}{2} \rfloor$. Then x is a coset leader.

Note:
Syndrome decoding is *not* efficient in general since the syndrome table is exponentially large. For an (n, k)-binary code, the syndrome table has size

$$2^{n-k}(n + (n - k)) = 2^{n-k}(2n - k) \text{ bits}$$

4

Golay codes

4.1 The (Binary) Golay Code

Let

$$\hat{B} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{12\times 11}$$

Let $\hat{G} = [I_{12} \mid \hat{B}]_{12\times 23}$. $\hat{G}$ is a GM for a $(23, 12)$-binary code called the (binary) Golay code C_{23}. We will prove $d(C_{23}) = 7$. **C_{23} is a perfect code.**

4.2 The Extended Golay Code C_{24}

C_{24} is the binary code with GM $G = [I_{12} \mid B]_{12\times 24}$, where

$$B = \left[\begin{array}{c|c} 0 & \\ 1 & \\ 1 & \hat{B} \\ \vdots & \\ 1 & \end{array}\right]_{12\times 12}$$

Properties of C_{24}:

- C_{24} is a $(24, 12)$-binary code.
- $GG^T = 0$. Hence $C_{24} \subseteq C_{24}^{\perp}$, so C_{24} is a **self-orthogonal code**.

  Since $C_{24} = C_{24}^{\perp}$. Hence C_{24} is a **self-dual code**.

- $B^T = B$ (so B is symmetric).
- A PCM for C_{24} is $H = [-B^T \mid I_{12}] = [B \mid I_{12}]$.
- Since $C_{24} = C_{24}^{\perp}$, H is also a GM for C_{24}.

Theorem 4.1

$d(C_{24}) = 8$.

Corollary 4.2

$d(C_{23}) = 7$.

4.3 A Decoding Algorithm For C_{24}

Recall $n = 24, k = 12, d = 8, e = 3$. $G = [I_{12} \mid B]$ and $H = [B \mid I_{12}]$ are both GMs and PCMs for C_{24}

Decoding strategy (IMLD) Compute a syndrome s of the received word r. Find a vector e of weight ≤ 3 that has the same syndrome. If no such e exists, reject r.

Correctness If the error vector has weight ≤ 3, then the decoder always makes the correct decision. If the error vector has weight > 3, the decoder will reject r or decode r to a codeword than the transmitted one.

There are 5 cases, not mutually exclusive, in the event $w(e) \leq 3$.

Suppose $r = (x, y)$ is received.

Algorithm 3: Decoding algorithm for C_{24}

```
Compute s_1 = [I_12 | B] r^T.
if s_1 = 0 then
    accept r and STOP // A
if w(s_1) ≤ 3 then
    e ← (s_1^T, 0)
    Decode r to c = r − e; STOP // B
foreach row i of B do
    if row i differs in one position (j), or two positions (j & k) with s_1 then
        Correct x in position j, or positions j, k;
        Correct y in position i; STOP // C
Compute s_2 = [B | I_12] r^T.
if w(s_2) ≤ 3 then
    e ← (0, s_2^T)
    Decode r to c = r − e; STOP // D
foreach row i of B do
    if row i differs in one position (j), or two positions (j & k) with s_2 then
        Correct x in position i;
        Correct y in position j, or positions j, k; STOP // E
Reject r
```

Decoding algorithm only needs B, no need for a syndrome table, which is much larger than B. Decoding is efficient and simple, which is good for hardware implementation.

Is C_{24} better than simpler codes such as replication codes or Hamming codes?

4.4 Reliability of C_{24}

- p = symbol error probability (BSC)
- $C = \{c_1, c_2, \ldots, c_M\}$
- w_i = probability that decoding algorithm makes an incorrect decision if c_i is sent.
- Error probability of C is $P_C = \frac{1}{M}\sum_{i=1}^{M} w_i = w_i$ = error probability of C
- $\boxed{1 - P_C = \text{Reliability of } C = \text{probability that } r \text{ is decoded correctly}}$

p	(1) $(1-p)^{12}$	(2) 1-$P_{C_{24}}$	(3) $1 - P_T$	(4) $1 - P_H$
0.1	0.282429	0.7857377	0.7112056	0.5490430
0.01	0.8863848	0.99990946	0.9964298	0.9903702
0.001	0.9880657	0.9999999895	0.99996402	0.998959
Rate	1	1/2	1/3	$11/15 \approx 0.73$

(1) If no source is used (no channel encoding is used), then the reliability for 12-bit messages is $(1-p)^{12}$.

(2) $w_i = 1 - \left[(1-p)^{24} + \binom{24}{1}p(1-p)^{23} + \binom{24}{2}p^2(1-p)^{22} + \binom{24}{3}(1-p)^{21}\right]$

$$P_{C_{24}} = \frac{1}{2^{12}}\sum_{i=1}^{2^{12}} w_i = w_i$$

(3) T = Triplication code

$$\underbrace{10110\ldots0}_{12} \to \underbrace{111\ 000\ 111\ 111\ 000\ldots\ 111}_{36}$$

$$1 - P_T = \left[(1-p)^3 + 3p(1-p)^2\right]^{12}$$

(4) $(15, 11)$-binary Hamming code

$$1 - P_H = (1-p)^{15} + 15p(1-p)^{14}$$

5

Cyclic codes

cyclic space

A subspace S of $V_n(F)$ is **cyclic** if $(a_0, a_1, \ldots, a_{n-1}) \in S \implies (a_{n-1}, a_0, a_1, \ldots, a_{n-2}) \in S$.

cyclic code

A **cyclic code** is a cyclic subspace of $V_n(F)$.

5.1 The Polynomial Ring $R = F[x]/(x^n - 1)$

Let $R = F[x]/(x^n - 1)$, where $F = \mathrm{GF}(q)$. Then R is a commutative ring (but not a field, since $x^n - 1$ is reducible over F).

We have the following bijection between $V_n(F)$ and R:

$$a = (a_0, a_1, a_2, \ldots, a_{n-1}) \longleftrightarrow a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}.$$

Vector addition and scalar multiplication preserved.

$a \cdot b$

Let $a, b \in V_n(F)$. Then $a \cdot b = c \in V_n(F)$, where $c \leftrightarrow c(x) = a(x) \cdot b(x) \bmod (x^n - 1)$.

ideal

Let R be a (finite) commutative ring. A non-empty subset I of R is an **ideal** of R if

(i) $a, b \in I \implies a + b \in I$;

(ii) $a \in I, b \in R \implies a \cdot b \in I$.

Theorem 5.1: algebraic characterization of cyclic subspaces of $V_n(F)$

Let S be a non-empty subset of $V_n(F)$. Let I be the associated polynomials in $R = F[x]/(x^n - 1)$. Then S is a cyclic subspace of $V_n(F)$ if and only if I is an ideal of R.

5.2 Ideals of $R = F[x]/(x^n - 1)$

ideal generated by g

Let R be a (commutative) ring, and let $g \in R$. Let $\langle g \rangle = \{g \cdot r : r \in R\}$. Then $\langle g \rangle$ is an ideal of R, called the **ideal generated by** g.

principal ideal

An ideal I of R is said to be **principal** if $I = \langle g \rangle$ for some $g \in I$.

principal ideal ring

A ring R is a **principal ideal ring** if every ideal of R is principal.

Theorem 5.2

$R = F[x]/(x^n - 1)$ is a principal ideal ring.

generator polynomial

Let I be an ideal of $R = F[x]/(x^n - 1)$.

- If $I = \{0\}$, then $x^n - 1$ is the **generator polynomial** of I.
- If $I \neq \{0\}$, then *the* monic polynomial of smallest degree in I is called the **generator polynomial** of I.

Theorem 5.3

Let I be a nonzero ideal of $R = F[x]/(x^n - 1)$.

1) There is a unique monic polynomial $g(x)$ of smallest degree in I; $I = \langle g \rangle$.
2) $g(x) \mid (x^n - 1)$ in $F[x]$.

Theorem 5.4

Let $h(x)$ be a monic divisor of $x^n - 1$ in $F[x]$. Then $h(x)$ is *the* generator polynomial of $\langle h(x) \rangle$.

Corollary 5.5

There is a 1-1 correspondence between ideals of R and monic divisors of $x^n - 1$, and thus also a 1-1 correspondence between cyclic subspaces of $V_n(F)$ and monic divisors of $x^n - 1$.

5.3 Dimension of a Cyclic Code

Theorem 5.6

Let $g(x)$ be a monic divisor of $x^n - 1$ over F, where $F = \mathrm{GF}(q)$. Suppose $\deg(g) = n - k$. Then the cyclic subspace S of $V_n(F)$ generated by $g(x)$ has dimension k.

5.4 GM of a Cyclic Code

Theorem 5.7

Let $g(x)$ be the generator polynomial of an (n,k)-cyclic code C over F (so $g(x)$ is a monic divisor of $x^n - 1$ over F of degree $n-k$). Then a (non-standard) GM for C is

$$G = \begin{bmatrix} g(x) \\ xg(x) \\ x^2 g(x) \\ \vdots \\ x^{k-1} g(x) \end{bmatrix}_{k \times n} .$$

Encoding Source messages are the polynomials in $F[x]$ of degree $< k$. If $m(x) = m_0 + m_1 x + \cdots + m_{k-1}x^{k-1}$, then encoding of m with respect to G is

$$c = \begin{bmatrix} m_0 & m_1 & \cdots & m_{k-1} \end{bmatrix} G = m_0 g(x) + m_1 x g(x) + \cdots + m_{k-1} x^{k-1} g(x) ,$$

so $c(x) = m(x)g(x)$. Note that no reduction by $x^n - 1$ is needed.

5.5 The Dual Code of a Cyclic Code

Let C be an (n,k)-cyclic code over F with generator polynomial $g(x)$. Let

$$g(x) = \underbrace{g_0}_{\neq 0} + g_1 x + \cdots + \underbrace{g_{n-k}}_{=1} x^{n-k} + \underbrace{g_{n-k+1}x^{n-k+1} + \cdots + g_{n-1}x^{n-1}}_{0} .$$

parity-check polynomial

The **parity-check polynomial** is $h(x) = (x^n - 1)/g(x)$.

Let

$$h(x) = \underbrace{h_0}_{\neq 0} + h_1 x + \cdots + \underbrace{h_k}_{=1} x^k + \underbrace{h_{k+1}x^{k+1} + \cdots + h_{n-1}x^{n-1}}_{0}$$

Also define $h_j = h_j \bmod n$ for all $j \in \mathbb{Z}$.

Observe that $g = (g_0, \ldots, g_{n-1})$ is orthogonal to the vector $\overline{h} = (h_{n-1}, \ldots, h_0)$ and all its cyclic shifts.

It follows that all cyclic shifts of g are orthogonal to all cyclic shifts of $\overline{h}$. Recall the GM for C, we

define H:

$$G = \begin{bmatrix} g_0 & g_1 & \cdots & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k-1} & g_{n-k} & \cdots & 0 \\ \vdots & \vdots & \ddots & & & & & \\ 0 & 0 & \cdots & g_0 & g_1 & \cdots & \cdots & g_{n-k} \end{bmatrix}_{k\times n}$$

$$H = \begin{bmatrix} h_k & h_{k-1} & \cdots & \cdots & h_0 & 0 & \cdots & 0 \\ 0 & h_k & h_{k-1} & \cdots & h_1 & h_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & & & & \\ 0 & 0 & \cdots & h_k & h_{k-1} & \cdots & \cdots & h_0 \end{bmatrix}_{(n-k)\times n}$$

We have $GH^T = 0$. Thus $C' \subseteq C^\perp$, where C' is the code generated by H. But $\text{rank}(H) = n-k$ (since $h_k = 1$), so $\dim(C') = n-k = \dim(C^\perp)$. Hence $C' = C^\perp$, and H is a (non-standard) PCM for C.

$C^\perp$ is cyclic

reciprocal polynomial

Let $h(x) = h_0 + h_1x + \cdots + h_kx^k$ be a polynomial of degree k (so $h_k \neq 0$). The **reciprocal polynomial** of $h(x)$ is $h_R(x) = x^k h(\frac{1}{x}) = h_k + h_{k-1}x + \cdots + h_0x^k$.

If $h_0 \neq 0$, we define $h^*(x) = h_0^{-1}h_R(x)$. So h^* is monic.

Theorem 5.8

Let C be an (n,k)-cyclic code over F with generator polynomial $g(x)$. Let $h(x) = (x^n-1)/g(x)$. Then $C^\perp$ is cyclic, with generator polynomial $h^*(x)$.

5.6 Computing Syndromes

Let C be an (n,k)-cyclic code over F with generator polynomial $g(x)$. We will find a "nice" PCM for C.

1. Find a GM for C of the form $[R \mid I_k]$

For $0 \leq i \leq k-1$, long division gives $x^{n-k+i} = \ell_i(x)g(x) + r_i(x)$, $\deg(r_i) < n-k$, $\deg(\ell_i) < k$. Then $x^{n-k+i} - r_i(x) = \ell_i(x)g(x) \in C$. Thus a GM for C is

$$G = \begin{bmatrix} -r_0(x) + x^{n-k} \\ -r_1(x) + x^{n-k+1} \\ \vdots \\ -r_{k-1}(x) + x^{n-1} \end{bmatrix}_{k\times n} = \left[\begin{array}{c|c} -x^{n-k} \pmod g(x) & \\ -x^{n-k+1} \pmod g(x) & \\ \vdots & I_k \\ -x^{n-1} \pmod g(x) & \end{array}\right] = [R \mid I_k]$$

Note that $\text{rank}(G) = k$.

2. A (systematic) PCM for C is $H = [I_{n-k} \mid -R^T]$

The rows of H^T (columns of H) are $x^0 \bmod g(x), x^1 \bmod g(x), \ldots, x^{n-1} \bmod g(x)$.

Theorem 5.9: computing syndromes

The syndrome of $r \in V_n(F)$ with respect to the above PCM is $s \in V_{n-k}(F)$, where

$$s(x) = r(x) \bmod g(x).$$

The syndromes of a vector and its cyclic shifts are closely related.

Theorem 5.10

Let $r(x)$ be a polynomial with syndrome polynomial $s(x) = s_0 + s_1 x + \cdots + s_{n-k-1}x^{n-k-1}$. The syndrome of $xr(x)$ is:

$$\begin{cases} xs(x), & \text{if } s_{n-k-1} = 0. \\ xs(x) - s_{n-k-1}g(x), & \text{if } s_{n-k-1} \neq 0. \end{cases}$$

Note that $xs(x)$ is not cyclic shift.

So given the syndrome s of r, we can easily compute the syndromes of cyclic shifts of r.

5.7 Burst Error Correcting

Cyclic codes are good for correcting burst errors.

cyclic burst length

Let $e \in V_n(F)$. The **cyclic burst length** of e is the length of the shortest *cyclic* block of e that contains all the nonzero components.

For example, the cyclic burst length of $e = \underline{0\,1\,1\,0\,1}\,0\,0\,0\,\underline{1\,0}$ is 7.

t-cyclic burst error correcting code

A linear code C is a **t-cyclic burst error correcting code** if all cyclic burst errors of length $\leq t$ are in different cosets of C, i.e., have different syndromes. The largest such t is the **cyclic burst error correcting capability** of C.

Theorem 5.11: bounds on burst error correcting capability

Let C be an (n,k,d)-code over GF(q). Let t be the cyclic burst error correcting capability of C. Then $\lfloor \frac{d-1}{2} \rfloor \leq t \leq (n-k)$.

In fact, we can prove that $t \leq \lfloor \frac{n-k}{2} \rfloor$.

5.8 Decoding Algorithm for Cyclic Burst Error Correcting Codes

Let C be an (n,k)-cyclic code over F with generator polynomial $g(x)$ and cyclic burst error correcting capability t (so $t \leq n-k$).

Let $r(x)$ be the received word. Let $s_i(x)$ denote the syndrome of $x^i r(x)$, $0 \leq i \leq n-1$.

Algorithm 4: Error-trapping decoding algorithm for cyclic burst error codes

```
for i ← 0 ... n − 1 do
    Compute s_i(x)
    if s_i has (non-cyclic) burst length ≤ t then
        e(x) ← x^{n−i}(s_i, 0)
        Decode r(x) to c(x) = r(x) − e(x)
Reject r.
```

5.9 Interleaving

Purpose Increase the cyclic burst error correcting capability of a code.

Let C be an (n,k)-code with cyclic burst error correcting capability t. Suppose

$$\begin{aligned} c_1 &= (c_{11}, c_{12}, \ldots, c_{1n}) \in C \\ c_2 &= (c_{21}, c_{22}, \ldots, c_{2n}) \in C \\ &\vdots \\ c_s &= (c_{s1}, c_{s2}, \ldots, c_{sn}) \in C \end{aligned}$$

Interleaving to a depth of s Instead of transmitting $c_1, c_2, \ldots, c_s$ in that order, transmit the *columns* of the above array:

$$c^* = (c_{11}, c_{21}, \ldots, c_{s1}, c_{12}, c_{22}, c_{22}, \ldots, c_{s2}, \ldots, c_{1n}, c_{2n}, \ldots, c_{sn})$$

Then, any cyclic burst of length $\leq st$ in c^* results in a cyclic burst of length $\leq t$ in each of the original codewords $c_1, c_2, \ldots, c_s$ (and these errors can be corrected).

Theorem 5.12: interleaving codes

Let C be an (n,k)-code over F with cyclic burst error capability t. Let C^* be the code obtained by interleaving C to a depth s.

1. C^* is an (ns, ks)-code over F with cyclic burst error correcting capability ts.
2. Suppose C is cyclic with generator polynomial $g(x)$. Then C^* is cyclic with generator polynomial $g(x^s)$.

6

BCH codes

6.1 Subfields and Extension fields

For any prime power q, $\mathrm{GF}(q)$ is a subfield of $\mathrm{GF}(q^m)$ and we can view $\mathrm{GF}(q^m)$ as an m-dimensional vector space over $\mathrm{GF}(q)$.

Example:

$\mathrm{GF}(2^{16})$ is a 16-dimensional vector space over $\mathrm{GF}(2)$.
$\mathrm{GF}(2^{16})$ is a 8-dimensional vector space over $\mathrm{GF}(2^2)$.
$\mathrm{GF}(2^{16})$ is a 4-dimensional vector space over $\mathrm{GF}(2^4)$.
$\mathrm{GF}(2^{16})$ is a 2-dimensional vector space over $\mathrm{GF}(2^8)$.
$\mathrm{GF}(2^{16})$ is a 1-dimensional vector space over $\mathrm{GF}(2^{16})$.

6.2 Minimal Polynomials

We call $\mathrm{GF}(q^m)$ the **extension field**, and $\mathrm{GF}(q)$ the subfield.

minimal polynomial of α over $\mathrm{GF}(q)$

Let $\alpha \in \mathrm{GF}(q^m)$. The **minimal polynomial of α over** $\mathrm{GF}(q)$, denoted $m_\alpha(y)$ is the monic polynomial of smallest degree in $\mathrm{GF}(q)[y]$ that α has a root.

Theorem 6.1

Let $\alpha \in \mathrm{GF}(q^m)$.

1. The minimal polynomial $m_\alpha(y)$ of α over $\mathrm{GF}(q)$ is unique.
2. $m_\alpha(y)$ is irreducible over $\mathrm{GF}(q)$.
3. $\deg(m_\alpha) \leq m$.
4. If $f \in \mathrm{GF}(q)[y]$, then $f(\alpha) = 0 \iff m_\alpha(y) \mid f(y)$.

6.3 Computing Minimal Polynomials

Theorem 6.2

Let $\alpha \in \mathrm{GF}(q^m)$. Then $\alpha \in \mathrm{GF}(q)$ if and only if $\alpha^q = \alpha$.

set of conjugates of α w.r.t. $\mathrm{GF}(q)$

Let $\alpha \in \mathrm{GF}(q^m)$. Let t be the smallest positive integer such that $\alpha^{q^t} = \alpha$ (note: $t \leq m$). Then the **set of conjugates of α w.r.t.** $\mathrm{GF}(q)$ is $C(\alpha) = \left\{\alpha, \alpha^q, \alpha^{q^2}, \ldots, \alpha^{q^{t-1}}\right\}$.

The t elements in $C(\alpha)$ are distinct.

Theorem 6.3

Let $\alpha \in \mathrm{GF}(q^m)$. Then the minimal polynomial of α over $\mathrm{GF}(q)$ is

$$m(y) = \prod_{\beta \in C(\alpha)} (y - \beta).$$

6.4 Factoring $x^n - 1$ over $\mathrm{GF}(q)$

Preliminaries Let p be the characteristic of $\mathrm{GF}(q)$. If $\gcd(n, q) \neq 1$, then write $n = \overline{n}p^\ell$, where $\ell \geq 1$ and $\gcd(\overline{n}, p) = 1$. Then $x^n - 1 = (x^{\overline{n}} - 1)^{p^\ell}$. So, WLOG, we shall assume that $\gcd(n, q) = 1$.

Let m be the smallest integer such that $q^m \equiv 1 \pmod{n}$, i.e., $n \mid (q^m - 1)$. Note that such an m exists.

Let α be a generator of $\mathrm{GF}(q^m)^*$. Let $\beta = \alpha^{(q^m-1)/n}$; note that $\beta \in \mathrm{GF}(q^m)$.

Also note, $\mathrm{ord}(\beta) = n$, and $1, \beta, \beta^2, \ldots, \beta^{n-1}$ are distinct. Furthermore, $(\beta^i)^n = (\beta^n)^i = 1$ for each $i \in [0, n-1]$. Hence $1, \beta, \beta^2, \ldots, \beta^{n-1}$ are roots of $x^n - 1$; and there aren't any other roots. So,

$$x^n - 1 = (x-1)(x-\beta)(x-\beta^2)\cdots(x-\beta^{n-1})$$

is the complete factorization of $x^n - 1$ over $\mathrm{GF}(q^m)$.

However, we seek the factorization of $x^n - 1$ over $\mathrm{GF}(q)$.

Consider β^i, where $0 \leq i \leq n-1$. Since β^i is a root of $x^n - 1$, we have $m_{\beta^i}(x) \mid (x^n - 1)$. Also, the roots of $m_{\beta^i}(x)$ are $C(\beta^i) = \left\{\beta^i, \beta^{iq}, \beta^{iq^2}, \ldots, \beta^{iq^{t-1}}\right\}$, where t is the smallest positive integer such that $iq^t \equiv i \pmod{n}$.

cyclotomic coset of $q \pmod{n}$ containing i

Suppose $\gcd(n, q) = 1$, and let $0 \leq i \leq n-1$. The **cyclotomic coset of** $q \pmod{n}$ **containing** i is

$$C_i = \{i, \quad iq \bmod n, \quad iq^2 \bmod n, \quad \ldots, \quad iq^{t-1} \bmod n\},$$

where t is the smallest positive integer such that $iq^t \equiv i \pmod{n}$. Also $C = \{C_i : 0 \leq i \leq n-1\}$ is the **set of cyclotomic cosets of** $q \bmod n$.

Theorem 6.4

Suppose $\gcd(n, q) = 1$.

- The number of monic irreducible factors of $x^n - 1$ over $\mathrm{GF}(q)$ is equal to the number of (distinct) cyclotomic cosets of q mod n.
- The number of monic irreducible factors of degree d is equal to the number of (distinct) cyclotomic cosets of q mod n of size d.

Theorem 6.5

Suppose $\gcd(n, q) = 1$. Let m be the smallest positive integer such that $q^m \pmod{n}$, and let $\beta \in \mathrm{GF}(q^m)$ be an element of order n. Then the monic irreducible factor of $x^n - 1$ over $\mathrm{GF}(q)$ are $\{m_{\beta^i}(x) : 0 \leq i \leq n-1\}$, where

$$m_{\beta^i}(x) = \prod_{j \in C_i} (x - \beta^j).$$

If $j \in C_i$, then $m_{\beta^j}(x) = m_{\beta^i}(x)$.

6.5 BCH Codes: Definition

BCH codes are cyclic codes that are constructed in such a way that a (useful) lower bound on their distance is known.

BCH code

A **BCH code** C over $\mathrm{GF}(q)$ of block length n and **designed distance** δ is a cyclic code generated by $g(x) = \mathrm{lcm}\{m_{\beta^i}(x) : a \leq i \leq a + \delta - 2\}$, for some integer a.

6.6 BCH Bound

Vanderdmonde matrix

A **Vanderdmonde matrix** over a field F is a matrix of the form

$$A(x_1, x_2, \ldots, x_t) = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_t \\ x_1^2 & x_2^2 & \cdots & x_t^2 \\ \vdots & \vdots & \vdots & \\ x_1^{t-1} & x_2^{t-1} & \cdots & x_t^{t-1} \end{bmatrix}_{t \times t},$$

where $x_1, x_2, \ldots, x_t \in F$.

Theorem 6.6

$\det(A(x_1, \ldots, x_t)) \neq 0$ if and only if $x_1, \ldots, x_t$ are distinct.

Corollary 6.7

A Vanderdmonde matrix $A(x_1, \ldots, x_t)$ is non-singular if and only if $x_1, x_2, \ldots, x_t$ are distinct.

Theorem 6.8: BCH bound

Let C be an (n,k)-BCH code over GF(q) with designed distance δ. Then $d(C) \geq \delta$.

6.7 BCH Decoding

We will present a decoding algorithm for one specific BCH code, named C_{15}. The decoding algorithm for C_{15} captures the essential ideas of a decoding algorithm for general BCH codes.

Let $q = 2, n = 15, m = 4$. Let GF$(2^4) = \mathbb{Z}_2[\alpha]/(\alpha^4 + \alpha + 1)$. Then α os a generator of GF$(2^4)^*$, and $\beta = \alpha$ has order 15.

Let

$$g(x) = m_\beta(x) \cdot m_{\beta^3}(x) = 1 + x + x^6 + x^7 + x^8$$

The roots of $g(x)$ include $\beta, \beta^2, \beta^3, \beta^4$, so $g(x)$ generates a $(15,7)$-BCH code C_{15} over GF(2) with $\delta = 5$. In fact, $d(C_{15}) = 5$, since $g(x)$ is a codeword of weight 5. This BCH code is called $C_{15} : (15,7,5)$-binary code. Note that C_{15} is a 2-error correcting code.

A PCM for C_{15} is

$$H = \begin{bmatrix} \beta^0 & \beta^1 & \beta^2 & \cdots & \beta^{14} \\ (\beta^3)^0 & (\beta^3)^1 & (\beta^3)^2 & \cdots & (\beta^3)^{14} \end{bmatrix}_{8 \times 15}$$

Note that H is a 2×15 matrix over GF(2^4). If we replace each element in H by its vector representation over GF(2), then we get an 8×15 matrix over GF(2).

The syndrome of $r \in V_{15}(\mathbb{Z}_2)$ is $Hr^T = \begin{bmatrix} r(\beta) \\ r(\beta^3) \end{bmatrix} \triangleq \begin{bmatrix} s_1 \\ s_3 \end{bmatrix}$.

So we don't need H to compute syndromes.

Decoding strategy If there is an error vector e of weight ≤ 2 that has the same syndrome (s_2, s_3) as r, then we decode r to $r - e$. Otherwise, we reject r.

Algorithm 5: Decoding algorithm for C_{15}

1. Received word is $r \in V_{15}(\mathbb{Z}_2)$.
2. Compute $s_1 = r(\beta)$ and $s_3 = r(\beta^3)$.
3. **if** $s_1 = s_3 = 0$ **then** accept r and STOP.
4. **if** $s_1^3 = s_3$ **then** correct r in position i where $s_1 = \beta^i$, and STOP.
5. **if** $s_1 = 0$ *(and* $s_3 \neq 0$*)* **then** reject r and STOP.
6. Form the *error locator polynomial* $r(z) = z^2 + s_1 z + \left(\frac{s_3}{s_1} + s_1^2\right)$, and find its roots in GF(2^4). If there are two (distinct) roots β^i and β^j, then correct r in positions i and j and STOP.
7. Reject.

The algorithm is guaranteed to make the correct decision if $w(e) \leq 2$.

More generally, suppose C is a binary (n,k)-BCH code with designed distance δ. Suppose the generator polynomial of C is $g(x) = \text{lcm}\{m_{\beta^i}(x) : i \in [\delta - 1]\}$ where $\beta \in$ GF(2^m) has order n. Then, $d(C) \geq \delta$. Let $t = \lfloor \frac{\delta - 1}{2} \rfloor$.

Suppose $c \in C$ is transmitted, $w(e) \leq t$, and r is received.

Compute $s_i = r(\beta^i)$ for each $1 \leq i \leq \delta - 1$, and form the *syndrome polynomial*:

$$s(z) = s_1 + s_2 z + s_3 z^3 + \cdots + s_{\delta-1} z^{\delta-2}$$

Fact From $s(z)$, the *error locator polynomial* can be efficiently computed. The roots of $\sigma(z)$ are β^{-j}, where j are the error positions.

Also, the algorithm generalizes to BCH codes over GF(q).

7

RS codes

7.1 Introduction

Reed-Solomon code

A **Reed-Solomon (RS) code** is a BCH code of length n over $\mathrm{GF}(q)$ where $n \mid (q-1)$.

Since $q^1 \equiv 1 \pmod{n}$, we have $m = 1$.

Suppose $n \mid (q-1)$, and let $\beta \in \mathrm{GF}(q)$ be an element of order n. Then $m_{\beta^i}(x) = x - \beta^i$ for all i.

A RS code C of length n over $\mathrm{GF}(q)$ with designed distance δ is a BCH code over $\mathrm{GF}(q)$ with generator polynomial

$$g(x) = (x-\beta^a)(x-\beta^{a+1})(x-\beta^{a+2})\cdots(x-\beta^{a+\delta-2})$$

for some a.

Since $\deg(g) = \delta - 1$, we have $w(g) \leq \delta$, so $d(c) \leq \delta$. $d(C) \geq \delta$ by BCH bound, hence $d(C) = \delta$.

Since $\dim(C) = k = n - \deg(g) = n - \delta + 1$, we have $k = n - d + 1$, so $d = n - k + 1$. Recall that $d \leq n - k + 1$ for any (n,k,d)-code. Thus, *RS are optimal* in the sense that, for any fixed n, k, q, they achieve maximum distance among all (n,k,d)-codes over $\mathrm{GF}(q)$.

7.2 RS Codes Have Good (Cyclic) Burst-Error Correcting Capability

Let C be a RS code of length n over $GF(2^r)$ and designed distance δ. Consider $c = (c_1, c_2, \ldots, c_n) \in C$, and let $e = \lfloor \frac{d-1}{2} \rfloor = \lfloor \frac{n-k}{2} \rfloor$. Note that $c_i \in GF(2^r)$.

By writing each c_i as a binary vector of length r, we can view c as a binary vector of length nr.

Now, if c is transmitted and if a cyclic burst error of length $\leq 1 + (e-1)r$ bits is introduced, then at most e symbols of c are received incorrectly. Thus, the received word can be decoded correctly.

Theorem 7.1

Let C be an (n,k)-RS code over $\mathrm{GF}(2^r)$. Then C', the code obtained by replacing each symbol in the codewords of C by the r-bit binary representations, is an (nr, kr)-binary code with cyclic burst error correcting capability $t = 1 + (e-1)r$ where $e = \lfloor \frac{n-k}{2} \rfloor$.

Example:

Consider $GF(2^8) = \mathbb{Z}_2[\alpha]/(\alpha^8 + \alpha^4 + \alpha^3 + \alpha^2 + 1)$.

Then $\beta = \alpha$ has order $n = 255$ (so $q = 256, n = 255$). Let

$$g(x) = \prod_{i=1}^{24}(x - \beta^i)$$

Then $g(x)$ is the generator polynomial for a $(255, 231, 25)$-RS code C with error correcting capability $e = 12$. The related code C' is a $(2040, 1848)$-binary code with cyclic burst error correcting capability $t = 89$.

The code C, and others derived from it, have widely been used in practice, including in CDs, DVDs, and QR codes.

8

Code-Based Cryptography

See Cameron's notes

9

Coding Theory 2

RS codes Optimal erasure codes (good for data storage). Efficient decoding algorithms + hardware architectures.

LDPC codes Low Density Parity Check codes. Good for soft-decision coding. Used in digital TV, optical communications, wireless communications, etc.

Concatenation codes BCH/RC + LDPC

Other codes Turbo codes, Raptor codes, Polar codes

List decoding Sudan-Guruswami algorithm. Lots of applications in theoretical computer science.

Index

O

P

R

S

T

V

W