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.

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/.

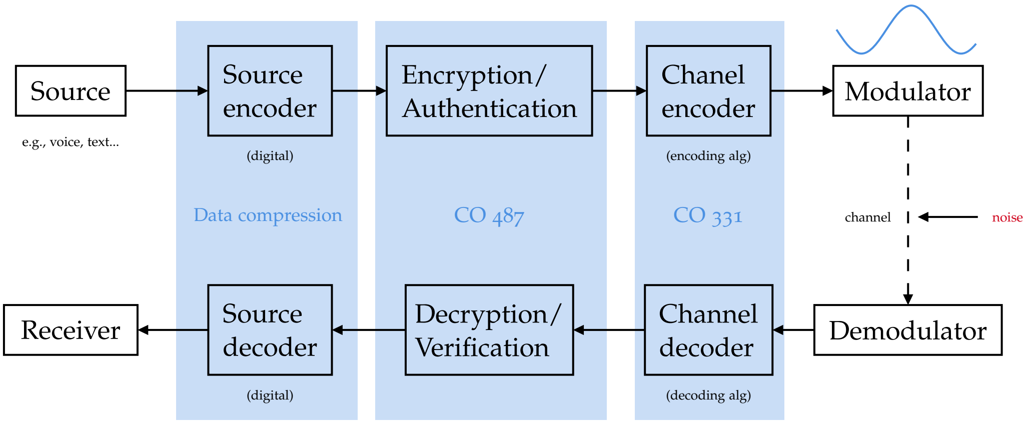
Sibelius Peng

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

		# err/codeword	# err/codeword	
Source msgs	Codeword	(always) detected	(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	→	00000
10	→	11100
01	→	00111
11	→	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:

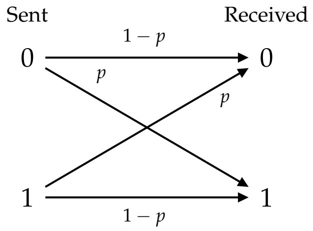
Let $A = \{a_1, \ldots, a_q\}$. Let $X_i =$ the i^{th} symbol sent. Let $Y_i =$ the i^{th} symbol received. Then for all $i \geq 1$, and all $i \leq j, k \leq q$,

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

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 $R = \log_q \frac{M}{n}$.

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):

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.

[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.

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 $C = \{\underset{\underset{c_1}{\uparrow}}{000}, \underset{\underset{c_2}{\uparrow}}{111}\}$. 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.

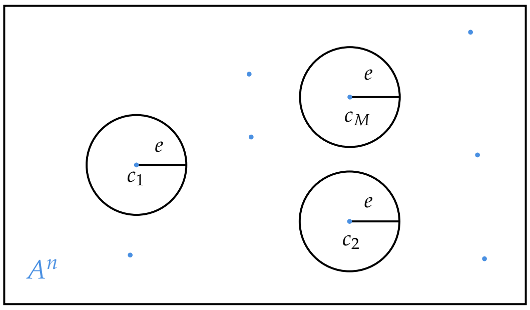
A natural question to ask is: given A, n, M, d, does there exist an $[n, M]$-code C over A of distance $\geq d$. This can be phrased as an equivalent sphere packing problem:

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.

Index