Coding Theory

CO 331

Prof. Alfred John 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 2020 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

0

Pre

Example: **Replication code**

source msgs		codewords
0	$\to$	0
1	$\to$	1

# of errors/codeword that be detected: 0
errors/codeword that can be corrected: 0
Rate: 1

---

source msgs		codewords
0	$\to$	00
1	$\to$	11

# of errors/codeword that be detected: 1
errors/codeword that can be corrected: 0
Rate: 1/2

---

source msgs		codewords
0	$\to$	000
1	$\to$	111

# of errors/codeword that be detected: 2
errors/codeword that can be corrected: 1 (nearest neighbour decoding)
Rate: 1/3

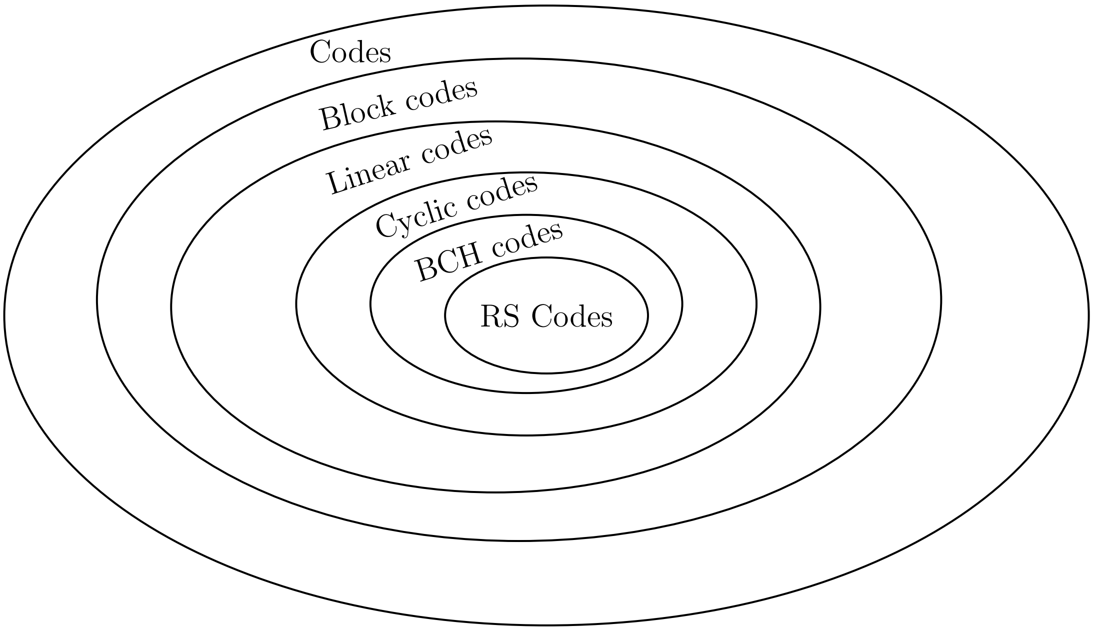
---

source msgs		codewords
0	$\to$	00000
1	$\to$	11111

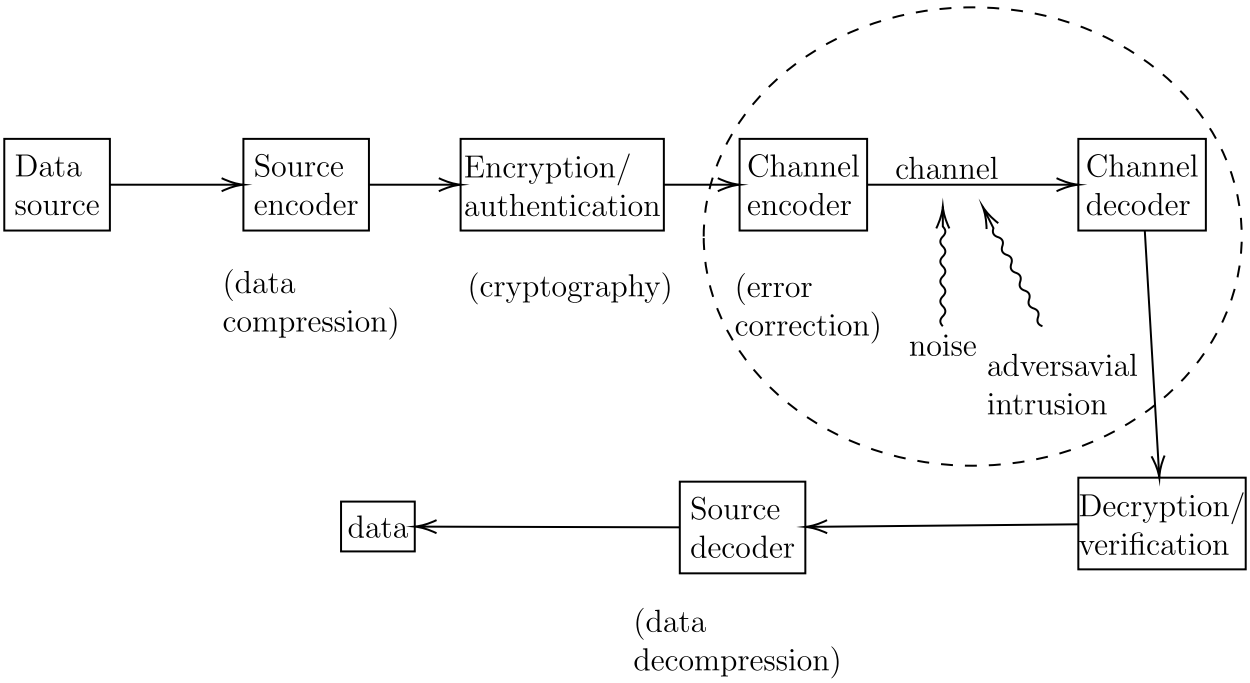
# of errors/codeword that be detected: 4
errors/codeword that can be corrected: 2 (nearest neighbour decoding)
Rate: 1/5

Goal of Coding Theory Design codes so that:

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



The big picture In its broadest sense, coding deals with the reliable, efficient, secure transmission of data over channels that are subject to inadvertent noise and malicious intrusion.



mid: Feb 26th

1 Introduction & Fundamentals

alphabet, word, length...

An *alphabet* A is a finite set of $q \geq 2$ symbols. E.g. $A = \{0, 1\}$.

A *word* is a finite sequence of symbols from A. (tuples or vectors)

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

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

A *codeword* is a word in C.

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

A block code C of length n containing M codewords over A is a subset $C \subseteq A^n$, with $|C| = M$. This is denoted by $[n, M]$.

Example:

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

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

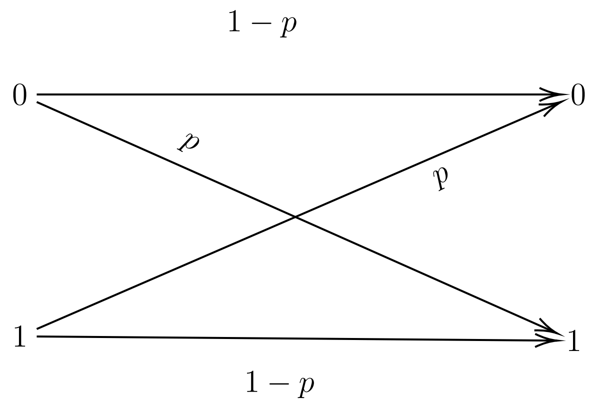
Encoding 1-1 map

The channel encoder transmits only codewords. But, what's received by the channel decoder might not be codeword.

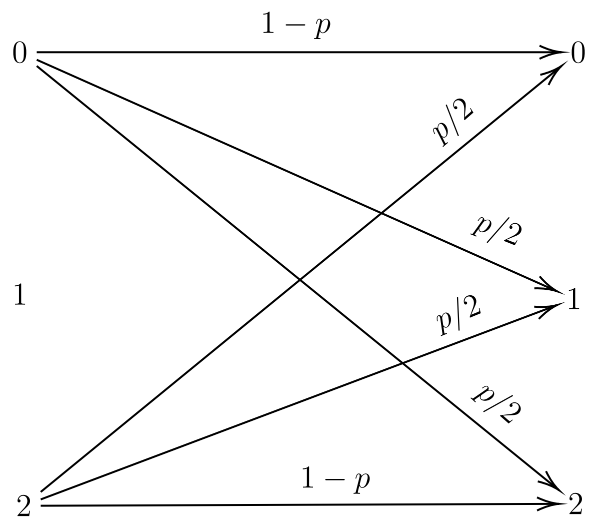
Example:

Suppose the channel decoder receives $r = 11001$. What should it do?

Example: $q = 2$ **(Binary symmetric channel, BSC)**



Example: $q = 3$



Assumptions about the communications channel

1) The channel only transmits symbols from A.

2) No symbols are deleted, added, or transposed.

3) (Errors are "random") Suppose the symbol transmitted are $X_1, X_2, X_3, \ldots$. Suppose the symbols received and $Y_1, Y_2, Y_3, \ldots$. Then for all $i \geq 1$, and all $i \leq j, k \leq q$,
$$Pr(Y_i = a_j | X_i = a_k) = \begin{cases} 1-p, & \text{if } j = k \\ \frac{p}{q-1}, & \text{if } j \neq k \end{cases}$$
where $p =$ symbol error prob.

Notes about BSC

(i) If $p = 0$, the channel is perfect.

(ii) If $p = \frac{1}{2}$, the channel is useless.

(iii) If $1 \geq p > \frac{1}{2}$, then simply flip all bits that are received.

(iv) WLOG, we will assume that $0 < p < \frac{1}{2}$.

(v) Analogously, for a q-ary channel, we can assume that $0 < p < \frac{q-1}{q}$. (Optional exercise)

Hamming distance

If $x, y \in A^n$, the *Hamming distance* $d(x, y)$ is the # of coordinate positions in which $x\&y$ differ.

The *distance of a code* C is

$$d(C) = \min\{d(x, y) \in C, x \neq y\}$$

Example:

$$d(10111, 01010) = 4$$

Theorem 1.1

d is a metric. For all $x, y, z \in A^n$

(i) $d(x, y) \geq 0$, and $d(x, y) = 0$ iff $x = y$.

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

(iii) $\triangle$ inequality $d(x, z) \leq d(x, y) + d(y, z)$

rate

The *rate* of an $[n, M]$-code C over A with $|A| = q$ is

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

If the source messages are all k-tuples over A,

$$R = \frac{\log_q(q^k)}{n} = \frac{k}{n}.$$

Example:

$$C = \{00000, 11100, 00111, 10101\} \qquad A = \{0, 1\}$$

Here $R = \frac{2}{5}$ and $d(C) = 2$.

1.1 Decoding Strategy

Let C be an $[n, M]$-code over A of distance d. Suppose some codeword is transmitted, and $r \in A^n$ is received. The channel decoder has to decide the following:

(i) no errors have occurred, accept r.

(ii) errors have occurred, and (decode) correct r to some codeword.

(iii) errors has occurred, correction is not possible.

1.1.1 Nearest Neighbour Decoding

Incomplete Maximum Likelihood Decoding (IMLD). Correct r to the unique codeword c for which $d(r, c)$ is smallest. If c is not unique, reject r. Complete MLD (CMLD). Same as IMLD, accept ties are broken arbitrarily.

Question Is IMLD a reasonable strategy?

Theorem 1.2

IMLD selects the codeword c that maximizes $P(r|c)$ prob. that r is received given that c was sent.

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

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

$\square$

The ideal strategy is to correct r to $c \in C$ that minimizes $P(c|r)$. This is Minimum error decoding (MED).

Example: **(IMD is not the same as MED)**

Let $C = \{\underbrace{000}_{c_1}, \underbrace{111}_{c_2}\}$. (corresponding to 0, 1).

Suppose $P(c_1) = 0.1, P(c_2) = 0.9$. Suppose $p = 1/4$ and $r = 100$.

IMLD $r \to 000$

MED

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

Similarly

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

So MED: $r \to 111$

Note

1. IMLD: Select c. s.t. $P(r|c)$ is maximum
   MED: Select c. s.t. $P(c|r)$ is maximum
2. MED has the drawback that it requires knowledge of $P(c_i)$, $1 \le i \le M$
3. Suppose source messages are equally likely, so $P(c_i) = \frac{1}{M}$, for each $1 \le i \le M$. Then

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

So IMLD is the same as MED.

4. In the remainder of the course, we will use IMLD/CMLD.

1.2 Error Correcting & Detecting Capabilities of a Code

- If C is used for error correction, the strategy is IMLD/CMLD.
- If C is used for error detection (only), the strategy is:

If $r \notin C$, then reject r; otherwise accept r.

e-error correcting code

A code C is called an *e-error correcting code* if the decoding always makes the correct decision if at most e errors per codeword are introduced. (Similarly: e-error detecting code)

Example:

$C = \{0000, 1111\}$ is 1-error correcting code, but not a 2-error correcting code.

$C = \{\underbrace{0\dots0}_{m}, \underbrace{1\dots1}_{m}\}$ is a $\lfloor \frac{m-1}{2} \rfloor$-error correcting code.

$C = \{0000, 1111\}$ is a 3-error detecting code.

Theorem 1.3

Suppose $d(C) = d$. Then C is a $(d-1)$-error detecting code.

Proof:

Suppose $c \in C$ is transmitted and r is received.

- If no error occur, then $r = c \in C$ and the decoder accepts r.
- If ≥ 1 and $\leq (d-1)$ errors occur, then $1 \leq d(r, c) \leq d - 1$. So, $r \notin C$, and hence the decoder rejects r.

$\square$

Theorem 1.4

If $d(C) = d$, then C is not a d-error detecting code.

Proof:

Since $d(C) = d$, there exist $c_1, c_2 \in C$ with $d(c_1, c_2) = d$. If c_1 is sent, it is possible that d errors occur and c_2 is received. In this case, the decoder accepts c_2. □

Theorem 1.5

If $d(C) = d$, then C is a $\lfloor \frac{d-1}{2} \rfloor$-error correcting code.

Proof:

Suppose $c \in C$ is transmitted, at most $\frac{d-1}{2}$ errors are introduced, and r is received. Let $c_1 \in C, c_1 \neq c$.

By $\triangle$ ineq, $d(c, c_1) \leq d(c, r) + d(r, c_1)$. So

$$d(r, c_1) \geq d(c, c_1) - d(c, r) \geq d - \frac{d-1}{2} = \frac{d+1}{2} \geq \frac{d-1}{2}$$

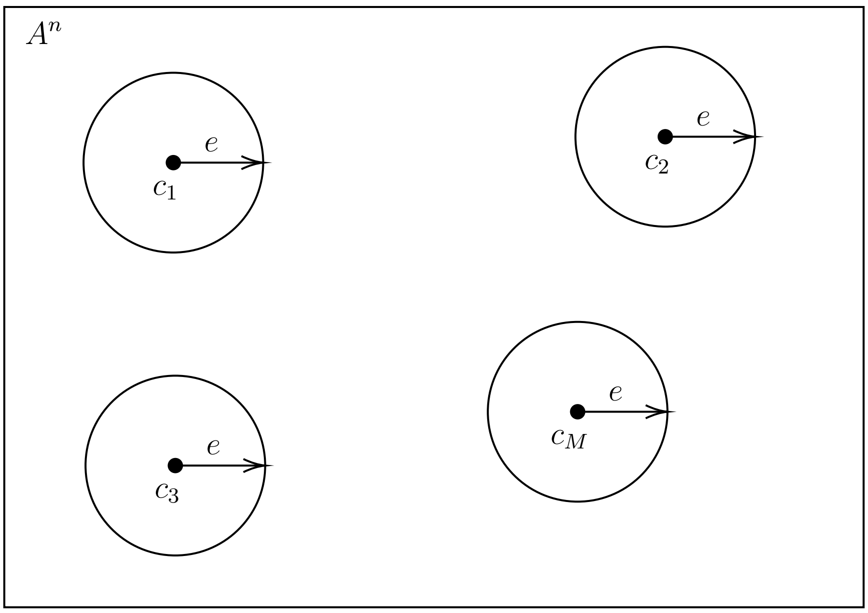
So c is the unique codeword closest to r.

So IMLD/CMLD will decode r to c. □

Theorem 1.6

If $d(C) = d$, then C is not a $\left(\lfloor \frac{d-1}{2} \rfloor + 1\right)$-error correcting code.

Question Given q, n, M, d, does there exist an $[n, M]$-code C over A (with $|A| = q$), with $d(C) = d$?



$C = \{c_1, c_2, \ldots, c_M\}$. Let $e = \lfloor \frac{d-1}{2} \rfloor$. For $c \in C$, let S_c =sphere of radius e centered at $c = \{r \in A^n : d(r,c) \leq e\}$. We proved: If $c_1, c_2 \in C, c_1 \neq c_2$, then $S_{c_1} \cap S_{c_2} \neq \varnothing$. The question can be viewed as a *sphere packing problem*: Can we place M spheres of radius e in A^n (such that no 2 spheres overlap)? This is purely combinatorial problem.

Example:

> Take $q = 2$, $n = 128$, $M = 2^{64}$, $d \geq 22$. Does a code with these parameters exist?
>
> **Answer** YES.

Question What are the codewords?

Question How do we encode and decode efficiently?

Preview 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.

Index