

# CO 331 - Coding Theory

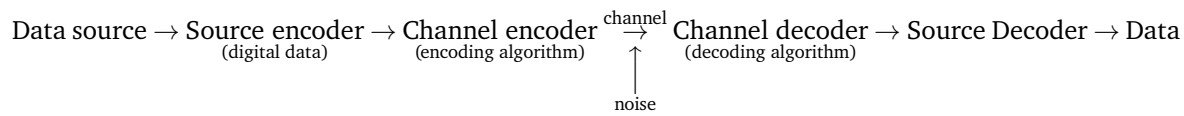
Cameron Roopnarine

Last updated: January 9, 2020

## Contents

<b>1</b>	<b>2020-01-06</b>	<b>2</b>
1.1	Example (Repetition Code) . . . . .	2
<b>2</b>	<b>2020-01-08</b>	<b>3</b>
2.1	Chapter 1: Introduction and Fundamentals . . . . .	3
2.2	Definition (Alphabet) . . . . .	3
2.3	Definition (Binary Alphabet) . . . . .	3
2.4	Definition (Word, Tuples, Vectors) . . . . .	3
2.5	Definition (Length) . . . . .	3
2.6	Definition (Code) . . . . .	3
2.7	Definition (codeword) . . . . .	3
2.8	Definition (Block code) . . . . .	3
2.9	Example (Block code) . . . . .	3
2.10	Example (Block code) . . . . .	4
2.11	Example . . . . .	4
2.12	Assumptions about the communications channel . . . . .	4
2.13	Example (Binary symmetric channel, BSC) . . . . .	4
2.14	Notes about BSC . . . . .	4
2.15	Definition (Hamming distance) . . . . .	4
2.16	Example (Hamming distance) . . . . .	5
2.17	Definition (Hamming distance of a code) . . . . .	5
2.18	Theorem . . . . .	5
2.19	Definition (Rate) . . . . .	5
2.20	Example (Rate) . . . . .	5

# 1 2020-01-06



## 1.1 Example (Repetition Code)

source message $\rightarrow$ codeword	# errors/codeword that can be detected	# errors/codeword that can be corrected	rate
0 $\rightarrow$ 0 1 $\rightarrow$ 1	0	0	1
0 $\rightarrow$ 00 1 $\rightarrow$ 11	1	0	$1/2$
0 $\rightarrow$ 000 1 $\rightarrow$ 111	2	1	$1/3$
0 $\rightarrow$ 00000 1 $\rightarrow$ 11111	4	2	$1/5$

### Goal of Coding Theory

Design codes so that:

- High information rate
- High error-correcting capability
- Efficient encoding/decoding algorithm

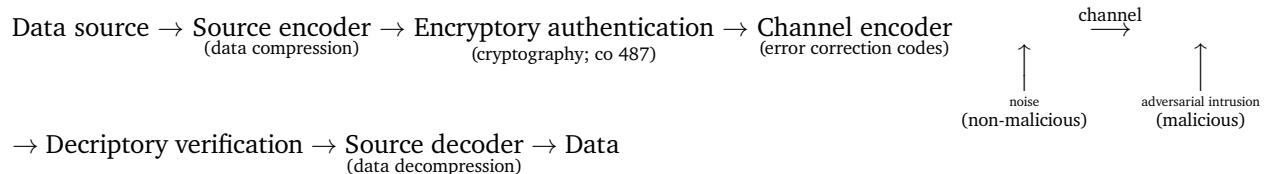
Codes  $\subset$  Block codes  $\subset$  Linear codes  $\subset$  Cyclic codes  $\subset$  BCH Codes  $\subset$  RS Codes

Requirements for this course:

- MATH 136
- Not required (but required to take the course): MATH 235
- Familiarity with: Groups, Fields, Ideals, Rings (these will be taught)
- Useful, if you have completed these you might be bored: PMATH 336, PMATH 334 [or the advanced equivalents]

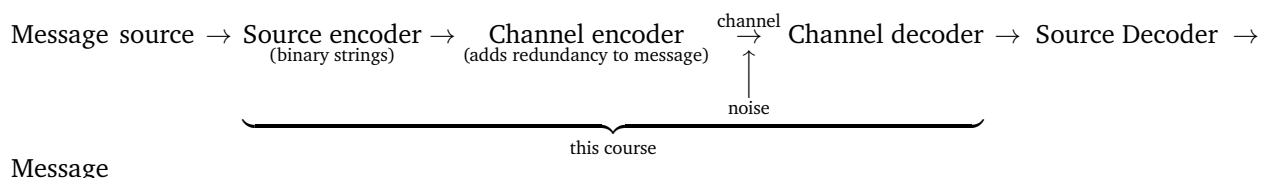
### The big picture

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



## 2 2020-01-08

### 2.1 Chapter 1: Introduction and Fundamentals



### 2.2 Definition (Alphabet)

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

Since we will be using the alphabet  $A = \{0, 1\}$  very often, we make the following definition.

### 2.3 Definition (Binary Alphabet)

The alphabet  $A = \{0, 1\}$  is a *binary alphabet*.

### 2.4 Definition (Word, Tuples, Vectors)

A *word* is a finite sequence of symbols from  $A$  (*tuples*, or *vectors*). We use the terms *vector* and *word* interchangeably for  $n$ -tuple.

### 2.5 Definition (Length)

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

### 2.6 Definition (Code)

A *code*  $C$  over  $A$  is a finite set of words over  $A$ . We define  $|C| \geq 2$ .

### 2.7 Definition (codeword)

A *codeword* is a word in  $C$ .

### 2.8 Definition (Block code)

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  $|C| = M$ . We refer to such a block code as an  $[n, M]$ -code over  $A$ .

### 2.9 Example (Block code)

Message $\rightarrow$ Codeword
00 $\rightarrow$ 00000
10 $\rightarrow$ 10110
01 $\rightarrow$ 01011
11 $\rightarrow$ 11101

The alphabet is  $A = \{0, 1\}$ . We have a 5-tuple since the length of each word is  $n = 5$ . The code is  $C = \{00000, 10110, 01011, 11101\}$ . Each element in  $C$  is a codeword, thus there are a total of 4 codewords.

## 2.10 Example (Block code)

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

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

This is an  $n$ -coding (1-1) map.

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

## 2.11 Example

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

## 2.12 Assumptions about the communications channel

- 1) Channels only transmit symbols from  $A$
- 2) No symbols are deleted, added, or transposed
- 3) (Errors are "random")

## 2.13 Example (Binary symmetric channel, BSC)

$q = 2$  (0 or 1)

Suppose that the symbols transmitted are  $X_1, X_2, X_3, \dots$ , and the symbols received are  $Y_1, Y_2, Y_3, \dots$ . Then for all  $i \geq 1$ , and all  $i \leq j, k \leq q$ ,

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

Here,  $p$  is the symbol error probability.

## 2.14 Notes about BSC

- (i) if  $p = 0$ , the channel is perfect
- (ii) if  $p = 1/2$ , the channel is useless
- (iii) if  $1/2 < p \leq 1$ , then simply flip all bits that aren't received
- (iv) WLOG, we'll assume that  $0 < p < 1/2$
- (v) Analogously, for any  $q$ -ary channel, we can assume that  $0 < p < \frac{q-1}{q}$

## 2.15 Definition (Hamming distance)

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

## 2.16 Example (Hamming distance)

The hamming distance of 10111 and 01010 is

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

## 2.17 Definition (Hamming distance of a code)

Let  $C$  be an  $[n, M]$ -code. The *Hamming distance*  $d$  of a code  $C$  is

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

## 2.18 Theorem

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

- (i)  $d(x, y) \geq 0$ , and  $d(x, y) = 0$  if and only if  $x = y$
- (ii)  $d(x, y) = d(y, x)$
- (iii) ( $\triangle$  inequality):  $d(x, z) \leq d(x, y) + d(y, z)$

## 2.19 Definition (Rate)

The *rate* (or *information rate*) of an  $[n, M]$ -code  $C$  over  $A$ , is

$$R = \frac{\log_q(M)}{n}$$

where  $q = |A|$ .

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

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

that is, there are  $q^k$  source messages.

## 2.20 Example (Rate)

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

We have a  $[2, 4]$ -code over  $\{0, 1\}$ .

$R = \frac{2}{5}$ , and  $d(C) = 2$  since 00111 and 10101 differ by 2 in the first and fourth bit.