# Coding and Cryptography

Dr T.A. Fisher

Michaelmas 2005

LaTeXed by Sebastian Pancratz

These notes are based on a course of lectures given by Dr T.A. Fisher in Part II of the Mathematical Tripos at the University of Cambridge in the academic year 2005–2006.

These notes have not been checked by Dr T.A. Fisher and should not be regarded as official notes for the course. In particular, the responsibility for any errors is mine — please email Sebastian Pancratz (`sfp25`) with any comments or corrections.

# Contents

# Introduction to Communication Channels

We model communication as illustrated in the following diagram.

$$\boxed{\text{Source}} \longrightarrow \boxed{\text{Encoder}} \;\underset{\uparrow\ \text{errors, ``noise''}}{\overset{\text{channel}}{\rightsquigarrow}}\; \boxed{\text{Decoder}} \longrightarrow \boxed{\text{Receiver}}$$

Examples include telegraphs, mobile phones, fax machines, modems, compact discs, or a space probe sending back a picture.

## Basic Problem

Given a source and a channel (modelled probabilistically), we must design an encoder and decoder to transmit messages economically (noiseless coding, data compression) and reliably (noisy coding).

**Example** (Noiseless coding)**.** In Morse code, common letters are given shorter codewords, e.g. A ·— , E · , Q ——·— , and Z ——·· .

Noiseless coding is adapted to the source.

**Example** (Noisy coding)**.** Every book has an ISBN $a_1\, a_2\, \ldots\, a_{10}$ where $a_i \in \{0, 1, \ldots, 9\}$ for $1 \le i \le 9$ and $a_{10} \in \{0, 1, \ldots, 9, X\}$ with $\sum_{j=1}^{10} j a_j \equiv 0 \pmod{11}$. This allows detection of errors such as

- one incorrect digit;
- transposition of two digits.

Noisy coding is adapted to the channel.

## Plan of the Course

I. Noiseless Coding
II. Error-control Codes
III. Shannon's Theorems
IV. Linear and Cyclic Codes
V. Cryptography

Useful books for this course include the following.

- D. Welsh: Codes & Cryptography, OUP 1988.
- C.M. Goldie, R.G.E. Pinch: Communication Theory, CUP 1991.
- T.M. Cover, J.A. Thomas: Elements of Information Theory, Wiley 1991.

- W. Trappe, L.C. Washington: Introduction to Cryptography with Coding Theory, Prentice Hall 2002.

The books mentioned above cover the following parts of the course.

| | W | G & P | C & T | T & W |
|---|---|---|---|---|
| I & III | ✓ | ✓ | ✓ | |
| II & IV | ✓ | ✓ | | ✓ |
| V | ✓ | | | ✓ |

# Overview

**Definition.** A *communication channel* accepts symbols from an *alphabet* $\Sigma_1 = \{a_1, \dots, a_r\}$ and it outputs symbols from an alphabet $\Sigma_2 = \{b_1, \dots, b_s\}$. The channel is modelled by the probabilities $\mathbb{P}(y_1 y_2 \dots y_n \text{ received}) \mid x_1 x_2 \dots x_n \text{ sent})$.

**Definition.** A *discrete memoryless channel (DMC)* is a channel with

$$p_{ij} = \mathbb{P}(b_j \text{ received} \mid a_i \text{ sent})$$

the same for each channel use and independent of all past and future uses of the channel. The *channel matrix* is $P = (p_{ij})$, an $r \times s$ stochastic matrix.

**Definition.** The *binary symmetric channel (BSC)* with error probability $0 \le p \le 1$ has $\Sigma_1 = \Sigma_2 = \{0, 1\}$. The channel matrix is $\left(\begin{smallmatrix} 1-p & p \\ p & 1-p \end{smallmatrix}\right)$. A symbol is transmitted with probability $1 - p$.

**Definition.** The *binary erasure channel* has $\Sigma_1 = \{0, 1\}$ and $\Sigma_2 = \{0, 1, \star\}$. The channel matrix is $\left(\begin{smallmatrix} 1-p & 0 & p \\ 0 & 1-p & p \end{smallmatrix}\right)$.

We model $n$ uses of a channel by the $n$th extension with input alphabet $\Sigma_1^n$ and ouput alphabet $\Sigma_2^n$.

A *code* $C$ of *length* $n$ is a function $\mathfrak{M} \to \Sigma_1^n$, where $\mathfrak{M}$ is the set of possible messages. Implicitly, we also have a decoding rule $\Sigma_2^n \to \mathfrak{M}$.

The *size* of $C$ is $m = |\mathfrak{M}|$. The *information rate* is $\rho(C) = \frac{1}{n} \log_2 m$. The *error rate* is $\hat{e}(C) = \max_{x \in \mathfrak{M}} \mathbb{P}(\text{error} \mid x \text{ sent})$.

**Definition.** A channel can *transmit reliably at rate* $R$ if there exists $(C_n)_{n=1}^{\infty}$ with $C_n$ a code of length $n$ such that

$$\lim_{n \to \infty} \rho(C_n) = R$$
$$\lim_{n \to \infty} \hat{e}(C_n) = 0$$

**Definition.** The *capacity* of a channel is the supremum over all reliable transmission rates.

**Fact.** A BSC with error probability $p < \frac{1}{2}$ has non-zero capacity.

# Chapter 1

# Noiseless Coding

**Notation.** For $\Sigma$ an alphabet, let $\Sigma^* = \bigcup_{n \geq 0} \Sigma^n$ be the set of all finite strings from $\Sigma$. Strings $x = x_1 \dots x_r$ and $y = y_1 \dots y_s$ have *concatenation* $xy = x_1 \dots x_r y_1 \dots y_s$.

**Definition.** Let $\Sigma_1, \Sigma_2$ be alphabets. A *code* is a function $f \colon \Sigma_1 \to \Sigma_2^*$. The strings $f(x)$ for $x \in \Sigma_1$ are called *codewords* or *words*.

**Example** (Greek Fire Code)**.** $\Sigma_1 = \{\alpha, \beta, \gamma, \dots, \omega\}$, $\Sigma_2 = \{1, 2, 3, 4, 5\}$. $f(\alpha) = 11, f(\beta) = 12, \dots, f(\psi) = 53, f(\omega) = 54$. Here, $xy$ means $x$ torches held up and another $y$ torches close-by.

**Example.** Let $\Sigma_1$ be the words in a given dictionary and $\Sigma_2 = \{A, B, C, \dots, Z, \text{␣}\}$. $f$ is "spell the word and follow by a space". We send a message $x_1 \dots x_n \in \Sigma_1^*$ as $f(x_1)f(x_2) \dots f(x_n) \in \Sigma_2^*$, i.e. $f$ extends to a function $f^* \colon \Sigma_1^* \to \Sigma_2^*$.

**Definition.** $f$ is *decipherable* if $f^*$ is injective, i.e. each string from $\Sigma_2$ corresponds to at most one message.

**Note 1.** Note that we need $f$ to be injective, but this is not enough.

**Example.** Let $\Sigma_1 = \{1, 2, 3, 4\}$, $\Sigma_2 = \{0, 1\}$ and $f(1) = 0, f(2) = 1, f(3) = 00, f(4) = 01$. Then $f^*(114) = 0001 = f^*(312)$. Here $f$ is injective but not decipherable.

**Notation.** If $|\Sigma_1| = m$, $|\Sigma_2| = a$ then $f$ is an $a$-ary code of size $m$.

Our aim is to construct decipherable codes with short word lengths. Assuming $f$ is injective, the following codes are always decipherable.

(i) A *block code* has all codewords the same length.
(ii) A *comma code* reserves a letter from $\Sigma_2$ to signal the end of a word.
(iii) A *prefix-free code* is one where no codeword is a prefix of any other distinct word. (If $x, y \in \Sigma_2^*$ then $x$ is a prefix of $y$ if $y = xz$ for some $z \in \Sigma_2^*$.)

**Note 2.** Note that (i) and (ii) are special cases of (iii). Prefix-free codes are sometimes called *instantaneous codes* or *self-punctuating codes.*

**Exercise 1.** Construct a decipherable code which is not prefix-free.

Take $\Sigma_1 = \{1, 2\}, \Sigma_2 = \{0, 1\}$ and set $f(1) = 0, f(2) = 01$.

**Theorem 1.1** (Kraft's Inequality)**.** Let $|\Sigma_1| = m, |\Sigma_2| = a$. A prefix-free code $f\colon \Sigma_1 \to \Sigma_2^*$ with word lengths $s_1, \dots, s_m$ exists if and only if

$$\sum_{i=1}^{m} a^{-s_i} \leq 1. \tag{$*$}$$

*Proof.* Rewrite ($*$) as

$$\sum_{l=1}^{s} n_l a^{-l} \leq 1 \tag{$**$}$$

where $n_l$ is the number of codewords of length $l$ and $s = \max_{1 \leq i \leq m} s_i$.

If $f\colon \Sigma_1 \to \Sigma_2^*$ is prefix-free then

$$n_1 a^{s-1} + n_2 a^{s-2} + \cdots + n_{s-1} a + n_s \leq a^s$$

since the LHS is the number of strings of length $s$ in $\Sigma_2$ with some codeword of $f$ as a prefix and the RHS is the number of strings of length $s$.

For the converse, given $n_1, \dots, n_s$ satisfying ($**$), we need to construct a prefix-free code $f$ with $n_l$ codewords of length $l$, for all $l \leq s$. We proceed by induction on $s$. The case $s = 1$ is clear. (Here, ($**$) gives $n_1 \leq a$, so we can choose a code.) By the induction hypothesis, there exists a prefix-free code $g$ with $n_l$ codewords of length $l$ for all $l \leq s-1$. ($**$) implies

$$n_1 a^{s-1} + n_2 a^{s-2} + \cdots + n_{s-1} a + n_s \leq a^s$$

where the first $s-1$ terms on the LHS sum to the number of strings of length $s$ with some codeword of $g$ as a prefix and the RHS is the number of strings of length $s$. Hence we can add at least $n_s$ new codewords of length $s$ to $g$ and maintain the prefix-free property. □

**Remark.** The proof is constructive, i.e. just choose codewords in order of increasing length, ensuring that no previous codeword is a prefix.

**Theorem 1.2** (McMillan)**.** Any decipherable code satisfies Kraft's inequality.

*Proof (Kamish).* Let $f\colon \Sigma_1 \to \Sigma_2^*$ be a decipherable code with word lengths $s_1, \dots, s_m$. Let $s = \max_{1 \leq i \leq m} s_i$. For $r \in \mathbb{N}$,

$$\left( \sum_{i=1}^{m} a^{-s_i} \right)^r = \sum_{l=1}^{rs} b_l a^{-l}$$

where

$$\begin{aligned} b_l &= |\{x \in \Sigma_1^r : f^*(x) \text{ has length } l\}| \\ &\leq |\Sigma_2^l| = a^l \end{aligned}$$

using that $f^*$ is injective. Then

$$\begin{aligned} \left( \sum_{i=1}^{m} a^{-s_i} \right)^r &\leq \sum_{l=1}^{rs} a^l a^{-l} = rs \\ \sum_{i=1}^{m} a^{-s_i} &\leq (rs)^{1/r} \to 1 \text{ as } r \to \infty. \end{aligned}$$

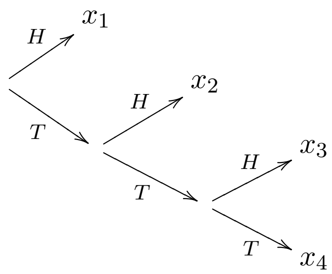
Therefore, $\sum_{i=1} a^{-s_i} \leq 1$. □

**Corollary 1.3.** A decipherable code with prescribed word lengths exists if and only if a prefix-free code with the same word lengths exists.

*Entropy* is a measure of "randomness" or "uncertainty". A random variable $X$ takes values $x_1, \ldots, x_n$ with probabilities $p_1, \ldots, p_n$, where $0 \leq p_i \leq 1$ and $\sum p_i = 1$. The entropy $H(X)$ is roughly speaking the expected number of fair coin tosses needed to simulate $X$.

**Example.** Suppose that $p_1 = p_2 = p_3 = p_4 = \frac{1}{4}$. Identify $\{x_1, x_2, x_3, x_4\} = \{HH, HT, TH, TT\}$, i.e. the entropy is $H = 2$.

**Example.** Let $(p_1, p_2, p_3, p_4) = (\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8})$.




Hence $H = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = \frac{7}{4}$.

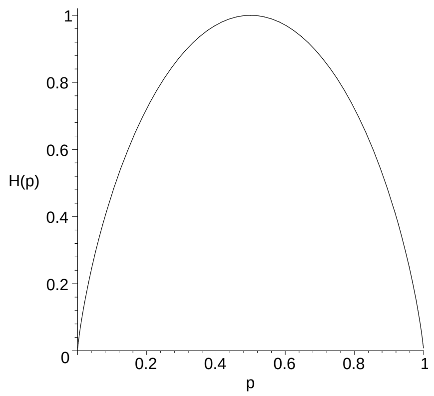
**Definition.** The *entropy* of $X$ is $H(X) = -\sum_{i=1}^{n} p_i \log p_i = H(p_1, \ldots, p_n)$, where in this course $\log = \log_2$.

**Note 3.** $H(X)$ is always non-negative. It is measured in bits.

**Exercise 2.** By convention $0 \log 0 = 0$. Show that $x \log x \to 0$ as $x \to 0$.

**Example.** A biased coin has $\mathbb{P}(\text{Heads}) = p$, $\mathbb{P}(\text{Tails}) = 1 - p$. We abbreviate $H(p, 1-p)$ as $H(p)$. Then

$$H(p) = -p \log p - (1-p) \log(1-p)$$
$$H'(p) = \log \frac{1-p}{p}$$



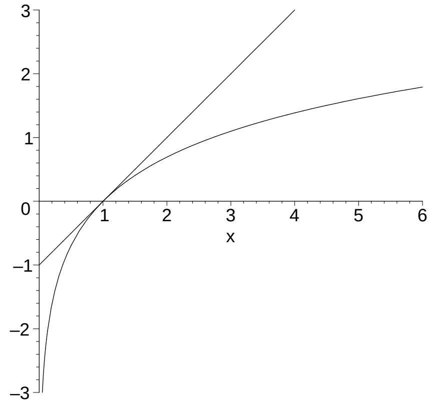

The entropy is greatest for $p = \frac{1}{2}$, i.e. a fair coin.

**Lemma 1.4** (Gibbs' Inequality)**.** Let $(p_1, \ldots, p_n)$ and $(q_1, \ldots, q_n)$ be probability distributions. Then

$$-\sum_{i=1}^{n} p_i \log p_i \leq -\sum_{i=1}^{n} p_i \log q_i$$

with equality if and only if $p_i = q_i$ for all $i$.

*Proof.* Since $\log x = \frac{\ln x}{\ln 2}$, we may replace log by ln for the duration of this proof. Let $I = \{1 \leq i \leq n : p_i \neq 0\}$.




We have

$$\ln x \leq x - 1 \quad \forall x > 0 \tag{$*$}$$

with equality if and only if $x = 1$. Hence

$$\begin{aligned} \ln \frac{q_i}{p_i} &\leq \frac{q_i}{p_i} - 1 \quad \forall i \in I \\ \therefore \quad \sum_{i \in I} p_i \ln \frac{q_i}{p_i} &\leq \sum_{i \in I} q_i - \sum_{i \in I} p_i \\ &= \sum_{i \in I} q_i - 1 \\ &\leq 0 \\ \therefore \quad -\sum_{i \in I} p_i \ln p_i &\leq -\sum_{i \in I} p_i \ln q_i \\ \therefore \quad -\sum_{i=1}^{n} p_i \ln p_i &\leq -\sum_{i=1}^{n} p_i \ln q_i \end{aligned}$$

If equality holds then $\sum_{i \in I} q_i = 1$ and $\frac{q_i}{p_i} = 1$ for all $i \in I$. Therefore, $p_i = q_i$ for all $1 \leq i \leq n$. □

**Corollary 1.5.** $H(p_1, \ldots, p_n) \leq \log n$ with equality if and only if $p_1 = \ldots = p_n = \frac{1}{n}$.

*Proof.* Take $q_1 = \cdots = q_n = \frac{1}{n}$ in Gibb's inequality. □

Let $\Sigma_1 = \{\mu_1, \ldots, \mu_m\}$, $|\Sigma_2| = a$. The random variable $X$ takes values $\mu_1, \ldots, \mu_m$ with probabilities $p_1, \ldots, p_m$.

**Definition.** A code $f\colon \Sigma_1 \to \Sigma_2^*$ is *optimal* if it is a decipherable code with the minimum possible expected word length $\sum_{i=1}^m p_i s_i$.

**Theorem 1.6** (Noiseless Coding Theorem)**.** The expected word length $\mathbb{E}(S)$ of an optimal code satisfies

$$\frac{H(X)}{\log a} \le \mathbb{E}(S) < \frac{H(X)}{\log a} + 1.$$

*Proof.* We first prove the lower bound. Take $f\colon \Sigma_1 \to \Sigma_2^*$ decipherable with word lengths $s_1, \dots, s_m$. Set $q_i = \frac{a^{-s_i}}{c}$ where $c = \sum_{i=1}^m a^{-s_i}$. Note $\sum_{i=1}^m q_i = 1$. By Gibbs' inequality,

$$\begin{aligned} H(X) &\le -\sum_{i=1}^m p_i \log q_i \\ &= -\sum_{i=1}^m p_i(-s_i \log a - \log c) \\ &= \left(\sum_{i=1}^m p_i s_i\right) \log a + \log c. \end{aligned}$$

By Theorem 1.2, $c \le 1$, so $\log c \le 0$.

$$\begin{aligned} \therefore \quad H(X) &\le \left(\sum_{i=1}^m p_i s_i\right) \log a \\ \therefore \quad \frac{H(X)}{\log a} &\le \mathbb{E}(S). \end{aligned}$$

We have equality if and only if $p_i = a^{-s_i}$ for some integers $s_1, \dots, s_m$.

For the upper bound, take $s_i = \lceil -\log_a p_i \rceil$. Then

$$\begin{aligned} -\log_a p_i &\le s_i \\ \therefore \quad \log_a p_i &\ge -s_i \\ \therefore \quad p_i &\ge a^{-s_i} \end{aligned}$$

Now $\sum_{i=1}^m a^{-s_i} \le \sum_{i=1}^m p_i = 1$. By Theorem 1.1, there exists a prefix-free code $f$ with word lengths $s_1, \dots, s_m$. $f$ has expected word length

$$\begin{aligned} \mathbb{E}(S) &= \sum_{i=1}^m p_i s_i \\ &< \sum_{i=1}^m p_i(-\log_a p_i + 1) \\ &= \frac{H(X)}{\log a} + 1. \end{aligned}$$ □

## Shannon–Fano Coding

This follows the above proof. Given $p_1, \dots, p_m$ set $s_i = \lceil -\log_a p_i \rceil$. Construct a prefix-free code with word lengths $s_1, \dots, s_m$ by choosing codewords in order of increasing length, ensuring that previous codewords are not prefixes.

**Example.** Let $a = 2, m = 5$.

| $i$ | $p_i$ | $\lceil -\log_2 p_i \rceil$ | |
|---|---|---|---|
| 1 | 0.4 | 2 | 00 |
| 2 | 0.2 | 3 | 010 |
| 3 | 0.2 | 3 | 011 |
| 4 | 0.1 | 4 | 1000 |
| 5 | 0.1 | 4 | 1001 |

We have $\mathbb{E}(S) = \sum p_i s_i = 2.8$. The entropy is $H = 2.121928\ldots$, so here $\frac{H}{\log a} = 2.121928\ldots$.

## Huffman Coding

For simplicity, let $a = 2$. Without loss of generality, $p_1 \geq \cdots \geq p_m$. The definition is recursive. If $m = 2$ take codewords 0 and 1. If $m > 2$, first take a Huffman code for messages $\mu_1, \ldots, \mu_{m-2}, \nu$ with probabilities $p_1, \ldots, p_{m-2}, p_{m-1} + p_m$. Then append 0 (resp. 1) to the codeword for $\nu$ to give a codeword for $\mu_{m-1}$ (resp. $\mu_m$).

**Remark.** (i) Huffman codes are prefix-free.
(ii) Exercise choice if some $p_j$ are equal.

**Example.** Consider the same case as in the previous example.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.4 | 1 | 0.4 | 1 | 0.4 | 1 | 0.6 | 0 |
| 0.2 | 01 | 0.2 | 01 | 0.4 | 00 | 0.4 | 1 |
| 0.2 | 000 | 0.2 | 000 | 0.2 | 01 | | |
| 0.1 | 0010 | 0.2 | 001 | | | | |
| 0.1 | 0011 | | | | | | |

We have $\mathbb{E}(S) = \sum p_i s_i = 2.2$.

**Theorem 1.7.** Huffman codes are optimal.

*Proof.* We show by induction on $m$ that Huffman codes of size $m$ are optimal.

If $m = 2$ the codewords are 0 and 1. This code is clearly optimal.

Assume $m > 2$. let $f_m$ be a Huffman code for $X_m$ which takes values $\mu_1, \ldots, \mu_m$ with probabilities $p_1 \geq \cdots \geq p_m$. $f_m$ is constructed from a Huffman code $f_{m-1}$ for $X_{m-1}$ which takes values $\mu_1, \ldots, \mu_{m-2}, \nu$ with probabilities $p_1, \ldots, p_{m-2}, p_{m-1} + p_m$. The expected word length is

$$\mathbb{E}(S_m) = \mathbb{E}(S_{m-1}) + p_{m-1} + p_m \tag{$*$}$$

Let $f'_m$ be an optimal code for $X_m$. Without loss of generality $f'_m$ is still prefix-free. By Lemma 1.8, without loss of generality the last two codewords of $f'_m$ have maximal length and differ only in the last digit. Say $f'_m(\mu_{m-1}) = y0, f'_m(\mu_m) = y1$ for some $y \in \{0, 1\}^*$. Let $f'_{m-1}$ be the prefix-free code for $X_{m-1}$ given by

$$f'_{m-1}(\mu_i) = f'_m(\mu_i) \quad \forall 1 \leq i \leq m-2$$

$$f'_{m-1}(\nu) = y.$$

The expected word length is

$$\mathbb{E}(S'_m) = \mathbb{E}(S'_{m-1}) + p_{m-1} + p_m \tag{$**$}$$

By the induction hypothesis, $f_{m-1}$ is optimal, hence $\mathbb{E}(S_{m-1}) \leq \mathbb{E}(S'_{m-1})$. So by ($*$) and ($**$), $\mathbb{E}(S_m) \leq \mathbb{E}(S'_m)$, so $f_m$ is optimal. □

**Lemma 1.8.** Suppose messages $\mu_1, \dots, \mu_m$ are sent with probabilities $p_1, \dots, p_m$. Let $f$ be an optimal prefix-free code with word lengths $s_1, \dots, s_m$.

(i) If $p_i > p_j$ then $s_i \leq s_j$.
(ii) Among all codewords of maximal lengths there exists two that differ only in the last digit.

*Proof.* If not, we modify $f$ by (i) swapping the $i$th and $j$th codewords, or (ii) deleting the last letter of each codeword of maximal length. The modified code is still prefix-free but has shorter expected word length, contradicting the optimality of $f$. □

## Joint Entropy

**Definition.** Let $X, Y$ be random variables that values in $\Sigma_1, \Sigma_2$.

$$H(X, Y) = -\sum_{x \in \Sigma_1} \sum_{y \in \Sigma_2} \mathbb{P}(X = x, Y = y) \log \mathbb{P}(X = x, Y = y)$$

This definition generalises to any finite number of random variables.

**Lemma 1.9.** Let $X, Y$ be random variables that values in $\Sigma_1, \Sigma_2$. Then

$$H(X, Y) \leq H(X) + H(Y)$$

with equality if and only if $X$ and $Y$ are independent.

*Proof.* Let $\Sigma_1 = \{x_1, \dots, x_m\}, \Sigma_2 = \{y_1, \dots, y_n\}$. Set

$$\begin{aligned} p_{ij} &= \mathbb{P}(X = x_i \wedge Y = y_j) \\ p_i &= \mathbb{P}(X = x_i) \\ q_i &= \mathbb{P}(Y = y_i) \end{aligned}$$

Apply Gibbs' inequality with probability distributions $\{p_{ij}\}$ and $\{p_i q_j\}$ to obtain

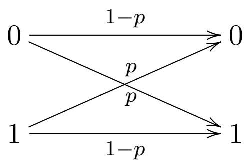
$$\begin{aligned} -\sum_{i,j} p_{ij} \log p_{ij} &\leq -\sum_{i,j} p_{ij} \log(p_i q_j) \\ &= -\sum_i \left( \sum_j p_{ij} \right) \log p_i - \sum_j \left( \sum_i p_{ij} \right) \log q_j \\ \therefore \quad H(X, Y) &\leq H(X) + H(Y) \end{aligned}$$

with equality if and only if $p_{ij} = p_i q_j$ for all $i, j$, i.e. if and only if $X, Y$ are independent. □

# Chapter 2

# Error-control Codes

**Definition.** A *binary $[n,m]$-code* is a subset $C \subset \{0,1\}^n$ of *size* $m = |C|$; $n$ is the *length* of the code, elements are called *codewords*. We use an $[n,m]$-code to send one of $m$ messages through a BSC making $n$ uses of the channel.




**Note 4.** Note $1 \leq m \leq 2^n$. Therefore, $0 \leq \frac{1}{n}\log m \leq 1$.

**Definition.** For $x, y \in \{0,1\}^n$ the *Hamming distance* is

$$d(x,y) = |\{i : 1 \leq i \leq n \wedge x_i \neq y_i\}|.$$

We consider three possible decoding rules.

(i) The *ideal observer* decoding rule decodes $x \in \{0,1\}^n$ as $c \in C$ maximising $\mathbb{P}(c \text{ sent} \mid x \text{ received})$.

(ii) The *maximum likelihood* decoding rule decodes $x \in \{0,1\}^n$ as $c \in C$ maximising $\mathbb{P}(x \text{ received} \mid c \text{ sent})$.

(iii) The *minimum distance* decoding rule decodes $x \in \{0,1\}^n$ as $c \in C$ minimising $d(x,c)$.

**Lemma 2.1.** If all messages are equally likely then (i) and (ii) agree.

**Lemma 2.2.** If $p < \frac{1}{2}$ then (ii) and (iii) agree.

**Remark.** The hypthesis of Lemma 2.1 is reasonable if we first carry out noiseless coding.

*Proof of Lemma 2.1.* By Bayes' rule,

$$\begin{aligned}\mathbb{P}(c \text{ sent} \mid x \text{ received}) &= \frac{\mathbb{P}(c \text{ sent and } x \text{ received})}{\mathbb{P}(x \text{ received})} \\ &= \frac{\mathbb{P}(c \text{ sent})}{\mathbb{P}(x \text{ received})}\,\mathbb{P}(x \text{ received} \mid c \text{ sent}).\end{aligned}$$

By the hypothesis, $\mathbb{P}(c \text{ sent})$ is independent of $c \in C$. So for fixed $x$, maximising $\mathbb{P}(c \text{ sent} \mid x \text{ received})$ is the same as maximising $\mathbb{P}(x \text{ received} \mid c \text{ sent})$. $\square$

*Proof of Lemma 2.2.* Let $r = d(x, c)$. Then

$$\mathbb{P}(x \text{ received} \mid c \text{ sent}) = p^r(1-p)^{n-r} = (1-p)^n \left(\frac{p}{1-p}\right)^r.$$

Since $p < \frac{1}{2}$, $\frac{p}{1-p} < 1$. So maximising $\mathbb{P}(x \text{ received} \mid c \text{ sent})$ is the same as minimising $d(x, c)$. □

**Example.** Codewords 000 and 111 are sent with probabilities $\alpha = \frac{9}{10}$ and $1 - \alpha = \frac{1}{10}$ through a BSC with error probability $p = \frac{1}{4}$. We receive 110.

$$\begin{aligned}
\mathbb{P}(000 \text{ sent} \mid 110 \text{ received}) &= \frac{\alpha p^2(1-p)}{\alpha p^2(1-p) + (1-\alpha)p(1-p)^2} \\
&= \frac{\alpha p}{\alpha p + (1-\alpha)(1-p)} \\
&= \frac{3}{4} \\
\mathbb{P}(111 \text{ sent} \mid 110 \text{ received}) &= \frac{1}{4}.
\end{aligned}$$

Therefore, the ideal observer decodes as 000. Maximimum likelihood and minimum distance rules both decode as 111.

From now on, we will use the minimum distance decoding rule.

**Remark.** (i) Minimum distance decoding may be expensive in terms of time and storage if $|C|$ is large.

(ii) We should specify a convention in the case of a tie, e.g. make a random choice, request to send again, etc.

We aim to detect, or even correct errors.

**Definition.** A code $C$ is

(i) *$d$-error detecting* if changing up to $d$ digits in each codeword can never produce another codeword.

(ii) *$e$-error correcting* if knowing that $x \in \{0, 1\}^n$ differs from a codeword in at most $e$ places, we can deduce the codeword.

**Example.** A *repetition code* of length $n$ has codewords $00\dots0$, $11\dots1$. This is a $[n, 2]$-code. It is $(n-1)$-error detecting and $\lfloor \frac{n-1}{2} \rfloor$-error correcting. But the information rate is only $\frac{1}{n}$.

**Example.** For the *simple parity check code*, also known as the paper tape code, we identify $\{0, 1\}$ with $\mathbb{F}_2$.

$$C = \{(c_1, \dots, c_n) \in \{0, 1\}^n : c_1 + \dots + c_n = 0\}.$$

This is a $[n, 2^{n-1}]$-code; it is 1-error detecting, but cannot correct errors. Its information rate is $\frac{n-1}{n}$.

We can work out the codeword of $0, \dots, 7$ by asking whether it is in $\{4, 5, 6, 7\}$, $\{2, 3, 6, 7\}$, $\{1, 3, 5, 7\}$ and setting the last bit to be the parity checker.

| | | | |
|---|---|---|---|
| 0 | 0000 | 4 | 1001 |
| 1 | 0011 | 5 | 1010 |
| 2 | 0101 | 6 | 1100 |
| 3 | 0110 | 7 | 1111 |

**Example.** Hamming's original [7,16]-code. Let $C \subset \mathbb{F}_2^7$ be defined by

$$
\begin{aligned}
c_1 + c_3 + c_5 + c_7 &= 0 \\
c_2 + c_3 + c_6 + c_7 &= 0 \\
c_4 + c_5 + c_6 + c_7 &= 0
\end{aligned}
$$

There is an arbitrary choice of $c_3, c_5, c_6, c_7$ but then $c_1, c_2, c_4$ are forced. Hence, $|C| = 2^4$ and the information rate is $\frac{1}{n} \log m = \frac{4}{7}$.

Suppose we receive $x \in \mathbb{F}_2^7$. We form the *syndrome* $z = (z_1, z_2, z_4)$ where

$$
\begin{aligned}
z_1 &= x_1 + x_3 + x_5 + x_7 \\
z_2 &= x_2 + x_3 + x_6 + x_7 \\
z_4 &= x_4 + x_5 + x_6 + x_7
\end{aligned}
$$

If $x \in C$ then $z = (0, 0, 0)$. If $d(x, c) = 1$ for some $c \in C$ then $x_i$ and $c_i$ differ for $i = z_1 + 2z_2 + 4z_4$. The code is 1-error correcting.

**Lemma 2.3.** The Hamming distance $d$ on $\mathbb{F}_2^n$ is a metric.

*Proof.* (i) $d(x, y) \geq 0$ with equality if and only if $x = y$.

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

(iii) Triangle inequality. Let $x, y, z \in \mathbb{F}_2^n$.

$$\{1 \leq i \leq n : x_i \neq z_i\} \subset \{1 \leq i \leq n : x_i \neq y_i\} \cup \{1 \leq i \leq n : y_i \neq z_i\}$$

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

**Remark.** We can also write $d(x, y) = \sum_{i=1}^n d_1(x_i, y_i)$ where $d_1$ is the discrete metric on $\mathbb{F}_2$.

**Definition.** The *minimum distance* of a code is the minimum value of $d(c_1, c_2)$ for $c_1, c_2$ distinct codewords.

**Lemma 2.4.** Let $C$ have minimum distance $d$.

(i) $C$ is $(d - 1)$-error detecting, but cannot detect all sets of $d$ errors.

(ii) $C$ is $\lfloor \frac{d-1}{2} \rfloor$-error correcting, but cannot correct all sets of $\lfloor \frac{d-1}{2} \rfloor + 1$ errors.

*Proof.* (i) $d(c_1, c_2) \geq d$ for all distinct $c_1, c_2 \in C$. Therefore, $C$ is $(d - 1)$-error detecting. But $d(c_1, c_2) = d$ for some $c_1, c_2 \in C$. Therefore, $C$ cannot correct all sets of $d$ errors.

(ii) The closed Hamming ball with centre $x \in \mathbb{F}_2^n$ and radius $r \geq 0$ is $B(x, r) = \{y \in \mathbb{F}_2^n : d(x, y) \leq r\}$. Recall, $C$ is $e$-error correcting if and only if

$$\forall \text{ distinct } c_1, c_2 \in C \quad B(c_1, e) \cap B(c_2, e) = \emptyset.$$

If $x \in B(c_1, e) \cap B_(c_2, e)$ then

$$\begin{aligned} d(c_1, c_2) &\le d(c_1, x) + d(x, c_2) \\ &\le 2e \end{aligned}$$

So if $d \ge 2e + 1$ then $C$ is $e$-error correcting. Take $e = \lfloor \frac{d-1}{2} \rfloor$.
Let $c_1, c_2 \in C$ with $d(c_1, c_2) = d$. Let $x \in \mathbb{F}_2^n$ differ from $c_1$ in $e$ digits where $c_1$ and $c_2$ differ too. Then $d(x, c_1) = e$, $d(x, c_2) = d - e$. If $d \le 2e$ then $B(c_1, e) \cap B(c_2, e) \neq \emptyset$, i.e. $C$ cannot correct all sets of $e$-errors. Take $e = \lceil \frac{d}{2} \rceil = \lfloor \frac{d-1}{2} \rfloor + 1$. □

**Notation.** An $[n, m]$-code with minimum distance $d$ is an $[n, m, d]$-code.

**Example.** (i) The repetition code of length $n$ is an $[n, 2, n]$-code.
(ii) The simple parity check code of length $n$ is an $[n, 2^{n-1}, 2]$-code.
(iii) Hamming's $[7, 16]$-code is 1-error correcting. Hence $d \ge 3$. Also, 0000000 and 1110000 are both codewords. Therefore, $d = 3$, and this code is a $[7, 16, 3]$-code. It is 2-error detecting.

## Bounds on Codes

**Notation.** Let $V(n, r) = |B(x, r)| = \sum_{i=0}^{r} \binom{n}{i}$, independently of $x \in \mathbb{F}_2^n$.

**Lemma 2.5** (Hamming's Bound)**.** An $e$-error correcting code $C$ of length $n$ has

$$|C| \le \frac{2^n}{V(n, e)}.$$

*Proof.* $C$ is $e$-error correcting, so $B(c_1, e) \cap B(c_2, e) = \emptyset$ for all distinct $c_1, c_2 \in C$. Therefore,

$$\begin{aligned} \sum_{c \in C} |B(c, e)| &\le |\mathbb{F}_2^n| = 2^n \\ \therefore \quad |C| V(n, e) &\le 2^n \end{aligned}$$

□

**Definition.** A code $C$ of length $n$ that can correct $e$-errors is *perfect* if

$$|C| = \frac{2^n}{V(n, e)}.$$

Equivalently, for all $x \in \mathbb{F}_2^n$ there exists a unique $c \in C$ such that $d(x, c) \le e$. Also equivalently, $\mathbb{F}_2^n = \bigcup_{c \in C} B(c, e)$, i.e. any $e + 1$ errors will make you decode wrongly.

**Example.** Hamming's $[7, 16, 3]$-code is $e = 1$ error correcting and

$$\frac{2^n}{V(n, e)} = \frac{2^7}{V(7, 1)} = \frac{2^7}{1 + 7} = 2^4 = |C|$$

i.e. this code is perfect.

**Remark.** If $\frac{2^n}{V(n,e)} \notin \mathbb{Z}$ then there does not exist a perfect $e$-error correcting code of length $n$. The converse is false (see Example Sheet 2 for the case $n = 90$, $e = 2$).

**Definition.** $A(n,d) = \max\{m : \text{there exists an } [n,m,d]\text{-code}\}$.

**Example.** We have

$$A(n,1) = 2^n \qquad A(n,n) = 2 \qquad A(n,2) = 2^{n-1}$$

In the last case, we have $A(n,2) \geq 2^{n-1}$ by the simple parity check code. Suppose $C$ has length $n$ and mininum distance 2. Let $\bar{C}$ be obtained from $C$ by switching the last digit of every codeword. Then $2|C| = |C \cup \bar{C}| \leq |\mathbb{F}_2^n| = 2^n$, so $A(n,2) = 2^{n-1}$.

**Lemma 2.6.** $A(n,d+1) \leq A(n,d)$.

*Proof.* Let $m = A(n,d+1)$ and pick $C$ with parameters $[n,m,d+1]$. Let $c_1, c_2 \in C$ with $d(c_1,c_2) = d+1$. Let $c_1'$ differ from $c_1$ in exactly one of the places where $c_1$ and $c_2$ differ. Hence $d(c_1',c_2) = d$. If $c \in C \setminus \{c_1\}$ then

$$\begin{aligned} & d(c,c_1) \leq d(c,c_1') + d(c_1',c_1) \\ \implies & d+1 \leq d(c,c_1') + 1 \\ \implies & d(c,c_1') \geq d. \end{aligned}$$

Replacing $c_1$ by $c_1'$ gives an $[n,m,d]$-code. Therefore, $m \leq A(n,d)$. $\square$

**Corollary 2.7.** Equivalently, we have

$$A(n,d) = \max\{m : \text{there exists an } [n,m,d']\text{-code for some } d' \geq d\}.$$

**Proposition 2.8.**

$$\frac{2^n}{V(n,d-1)} \leq A(n,d) \leq \frac{2^n}{V(n,\lfloor\frac{d-1}{2}\rfloor)}$$

The lower bound is known as the Gilbert Shannon Varshanov (GSV) bound or sphere covering bound. The upper bound is known as Hamming's bound or sphere packing bound.

*Proof of the GSV bound.* Let $m = A(n,d)$. Let $C$ be an $[n,m,d]$-code. Then there does not exist $x \in \mathbb{F}_2^n$ with $d(x,c) \geq d \; \forall c \in C$, otherwise we could replace $C$ by $C \cup \{x\}$ to get an $[n,m+1,d]$-code. Therefore,

$$\begin{aligned} \mathbb{F}_2^n &= \bigcup_{c \in C} B(c,d-1) \\ \therefore \quad 2^n &\leq \sum_{c \in C} |B(c,d-1)| = mV(n,d-1). \end{aligned}$$

$\square$

**Example.** Let $n = 10, d = 3$. We have $V(n,2) = 56$, $V(n,1) = 11$.

$$\begin{aligned} \frac{2^{10}}{56} &\leq A(10,3) \leq \frac{2^{10}}{11} \\ \therefore \quad 19 &\leq A(10,3) \leq 93. \end{aligned}$$

It is known that $72 \leq A(10,3) \leq 79$, but the exact value is not known.

We study $\frac{1}{n}\log A(n,\lfloor n\delta\rfloor)$ as $n \to \infty$ to see how large the information rate can be for a given error rate.

**Proposition 2.9.** Let $0 < \delta < \frac{1}{2}$. Then

(i) $\log V(n, \lfloor n\delta \rfloor) \leq nH(\delta)$
(ii) $\frac{1}{n} \log A(n, \lfloor n\delta \rfloor) \geq 1 - H(\delta)$

where $H(\delta) = -\delta \log \delta - (1-\delta)\log(1-\delta)$ as before.

*Proof.* We first show that (i) implies (ii). By the GSV bound,

$$\begin{aligned} A(n, \lfloor n\delta \rfloor) &\geq \frac{2^n}{V(n, \lfloor n\delta \rfloor)} \\ \therefore \quad \frac{\log A(n, \lfloor n\delta \rfloor)}{n} &\geq 1 - \frac{\log V(n, \lfloor n\delta \rfloor)}{n} \\ &\geq 1 - H(\delta). \end{aligned}$$

Now we prove (i). Since $H(\delta)$ is increasing for $\delta \leq \frac{1}{2}$, we may assume $n\delta \in \mathbb{Z}$.

$$\begin{aligned} 1 &= (\delta + (1-\delta))^n \\ &= \sum_{i=0}^{n} \binom{n}{i} \delta^i (1-\delta)^{n-i} \\ &\geq \sum_{i=0}^{n\delta} \binom{n}{i} \delta^i (1-\delta)^{n-i} \\ &= (1-\delta)^n \sum_{i=0}^{n\delta} \binom{n}{i} \left(\frac{\delta}{1-\delta}\right)^i \\ &\geq (1-\delta)^n \sum_{i=0}^{n\delta} \binom{n}{i} \left(\frac{\delta}{1-\delta}\right)^{n\delta} \\ &= \delta^{n\delta}(1-\delta)^{n(1-\delta)} V(n, n\delta) \end{aligned}$$

Take logarithms to obtain

$$\begin{aligned} & 0 \geq n\delta \log \delta + n(1-\delta)\log(1-\delta) + \log V(n, n\delta) \\ \therefore \quad & 0 \geq -nH(\delta) + \log V(n, n\delta) \end{aligned}$$ □

In fact, the constant $H(\delta)$ is in Proposition 2.9 (i) is best possible.

**Lemma 2.10.**

$$\lim_{n \to \infty} \frac{V(n, \lfloor n\delta \rfloor)}{n} = H(\delta).$$

*Proof.* Without loss of generality assume $0 < \delta < \frac{1}{2}$. Let $0 \leq r \leq \frac{n}{2}$ and recall $V(n, r) = \sum_{i=0}^{r} \binom{n}{i}$. Therefore,

$$\binom{n}{r} \leq V(n, r) \leq (r+1)\binom{n}{r} \tag{$*$}$$

Stirling's formula states

$$\ln n! = n \ln n - n + \mathcal{O}(\log n)$$

$$\begin{aligned}
\therefore \quad \ln\binom{n}{r} &= (n\ln n - n) - (r\ln r - r) - ((n-r)\ln(n-r) - (n-r)) \\
&\quad + \mathcal{O}(\log n) \\
\therefore \quad \log\binom{n}{r} &= -r\log\frac{r}{n} - (n-r)\log\frac{n-r}{n} + \mathcal{O}(\log n) \\
&= nH\left(\frac{r}{n}\right) + \mathcal{O}(\log n)
\end{aligned}$$

By $(*)$,

$$H\left(\frac{r}{n}\right) + \mathcal{O}\left(\frac{\log n}{n}\right) \le \frac{\log V(n,r)}{n} \le H\left(\frac{r}{n}\right) + \mathcal{O}\left(\frac{\log n}{n}\right)$$
$$\therefore \quad \lim_{n\to\infty} \frac{\log V(n, \lfloor n\delta \rfloor)}{n} = H(\delta)$$

If $\frac{1}{2} \le \delta$, we can use the symmetry of the binomial coefficients and the entropy to swap $\delta$ and $1-\delta$. □

# New Codes from Old

Let $C$ be an $[n, m, d]$-code.

(i) The *parity check extension* of $C$ is

$$\bar{C} = \{(c_1, \dots, c_n, \sum_{i=1}^{n} c_i : (c_1, \dots, c_n) \in C\},$$

where the sum is modulo 2.

(ii) Fix $1 \le i \le n$. Deleting the $i$th digit from each codeword gives a *punctured code*, with (assuming $d \ge 2$) parameters $[n-1, m, d']$ where $d - 1 \le d' \le d$.

(iii) Fix $1 \le i \le n$, $a \in \mathbb{F}_2$. The *shortened code* is

$$\{(c_1, \dots, c_{i-1}, c_{i+1}, \dots, c_n) : (c_1, \dots, c_{i-1}, a, c_{i+1}, \dots, c_n) \in C\}$$

It has parameters $[n-1, m', d']$ with $d' \ge d$ and $m' \ge \frac{m}{2}$ for a suitable choice of $a$.

# Chapter 3

# Shannon's Theorems

**Definition.** A *source* is a sequence of random variables $X_1, X_2, \dots$ taking values in some alphabet $\Sigma$. A source $X_1, X_2, \dots$ is *Bernoulli*, or *memoryless*, if $X_1, X_2, \dots$ are independent identically distributed.

**Definition.** A source $X_1, X_2, \dots$ is *reliably encodeable at rate* $r$ if there exists subsets $A_n \subset \Sigma^n$ such that

(i) $\lim_{n\to\infty} \frac{\log|A_n|}{n} = r$;

(ii) $\lim_{n\to\infty} \mathbb{P}\big((X_1, \dots, X_n) \in A_n\big) = 1$.

**Definition.** The *information rate* $H$ of a source is the infimum of all reliable encoding rates.

**Note 5.** Note that $0 \le H \le \log|\Sigma|$.

Shannon's First Coding Theorem computes the information rate of certain sources, including Bernoulli sources.

Consider the probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Recall that a random variable $X$ is a function defined on $\Omega$ with some range, e.g. $\mathbb{R}$, $\mathbb{R}^n$, or $\Sigma$. We have a probability mass function

$$\begin{aligned} p_X \colon \Sigma &\to [0,1] \\ x &\mapsto \mathbb{P}(X = x) \end{aligned}$$

We consider

$$\begin{aligned} p(X) \colon \Omega \xrightarrow{X} \Sigma &\xrightarrow{\quad p_X \quad} [0,1] \\ \omega \longmapsto\!\!\!\!\!\!\!\!\quad &\longrightarrow \mathbb{P}(X = X(\omega)) \end{aligned}$$

Note that $p(X)$ is another random variable.

Recall that a sequence of random variables $X_1, X_2, \dots$ converges in probability to $c \in \mathbb{R}$ if

$$\forall \varepsilon > 0 \quad \lim_{n\to\infty} \mathbb{P}(|X_n - c| > \varepsilon) = 0.$$

We write this as

$$X_n \xrightarrow{\mathbb{P}} c \text{ as } n \to \infty.$$

**Fact** (Weak Law of Large Numbers, WLLN)**.** Let $X_1, X_2, \dots$ be i.i.d. real-valued random variables with finite expected value $\mu$. Then

$$\frac{1}{n}\sum_{i=1}^{n} X_i \xrightarrow{\mathbb{P}} \mu \text{ as } n \to \infty.$$

**Lemma 3.1.** The information rate of a Bernoulli source $X_1, X_2, \ldots$ is atmost the expected word length of an optimal code $f \colon \Sigma \to \{0,1\}^*$ for $X$.

*Proof.* Let $S_1, S_2, \ldots$ be the lengths of codewords when we encode $X_1, X_2, \ldots$ using $f$. Let $\varepsilon > 0$ and set

$$A_n = \{x \in \Sigma^n : f^*(x) \text{ has length less than } n(\mathbb{E}\, S_1 + \varepsilon)\}$$

Then

$$\begin{aligned}
\mathbb{P}((X_1, \ldots, X_n) \in A_n) &= \mathbb{P}\left( (\sum_{i=1}^{n} S_i < n(\mathbb{E}\, S_1 + \varepsilon) \right) \\
&= \mathbb{P}\left( |\frac{1}{n}\sum_{i=1}^{n} S_i - \mathbb{E}\, S_1| < \varepsilon \right) \\
&\to 1 \text{ as } n \to \infty
\end{aligned}$$

$f$ is decipherable, so $f^*$ is injective. Hence $|A_n| \leq 2^{n(\mathbb{E}\, S_1 + \varepsilon)}$. Making $A_n$ larger, we may assume $|A_n| = \lfloor 2^{n(\mathbb{E}\, S_1 + \varepsilon)} \rfloor$, so

$$\frac{\log|A_n|}{n} \to \mathbb{E}\, S_1 + \varepsilon$$

Therefore, $X_1, X_2, \ldots$ is reliably encodeable at rate $\mathbb{E}\, S_1 + \varepsilon$ for all $\varepsilon > 0$, so the information rate is at most $\mathbb{E}\, S_1$. □

**Corollary 3.2.** From Lemma 3.1 and the Noiseless Coding Theorem, a Bernoulli source $X_1, X_2, \ldots$ has information rate less than $H(X_1) + 1$.

Suppose we encode $X_1, X_2, \ldots$ in blocks

$$\underbrace{X_1, \ldots, X_N}_{Y_1}, \underbrace{X_{N+1}, \ldots, X_{2N}}_{Y_2}, \ldots$$

such that $Y_1, Y_2, \ldots$ take values in $\Sigma^N$.

**Exercise 3.** If $X_1, X_2, \ldots$ has information rate $H$ then $Y_1, Y_2, \ldots$ has information rate $NH$.

**Proposition 3.3.** The information rate $H$ of a Bernoulli source $X_1, X_2, \ldots$ is at most $H(X_1)$.

*Proof.* We apply the previous corollary to $Y_1, Y_2, \ldots$ and obtain

$$\begin{aligned}
NH &\leq H(Y_1) + 1 \\
&= H(X_1, \ldots, X_N) + 1 \\
&= \sum_{i=1}^{N} H(X_i) + 1 \\
&= NH(X_1) + 1 \\
\therefore \quad H &< H(X_1) + \frac{1}{N}
\end{aligned}$$

But $N \geq 1$ is arbitrary, so $H \leq H(X_1)$. □

**Definition.** A source $X_1, X_2, \ldots$ satisfies the *Asymptotic Equipartition Property* (AEP) for constant $H \geq 0$ if

$$-\frac{1}{n} \log p(X_1, \ldots, X_n) \to H \text{ as } n \to \infty.$$

**Example.** We toss a biased coin, $\mathbb{P}(\text{Heads}) = \frac{2}{3}$, $\mathbb{P}(\text{Tails}) = \frac{1}{3}$, 300 times. Typically we get about 200 heads and 100 tails. Each such sequence occurs with probability approximately $(\frac{2}{3})^{200}(\frac{1}{3})^{100}$.

**Lemma 3.4.** The AEP for a source $X_1, X_2, \ldots$ is equivalent to the following property:

$$\begin{aligned} &\forall \varepsilon > 0 \quad \exists n_0(\varepsilon) \quad \forall n \geq n_0(\varepsilon) \quad \exists T_n \subset \Sigma^n \text{ such that} \\ &\quad \text{(i) } \mathbb{P}\big((X_1, \ldots X_n) \in T_n\big) > 1 - \varepsilon \\ &\quad \text{(ii) } \forall (x_1, \ldots, x_n) \in T_n \quad 2^{-n(H+\varepsilon)} \leq p(x_n, \ldots, x_n) \leq 2^{-n(H-\varepsilon)} \end{aligned} \tag{$*$}$$

The $T_n$ are called *typical sets*.

*Proof.* If $(x_1, \ldots, x_n) \in \Sigma^n$ then we have the following equivalence

$$\begin{aligned} &2^{-n(H+\varepsilon)} \leq p(x_1, \ldots, x_n) \leq 2^{-n(H-\varepsilon)} \\ \iff \quad &|-\tfrac{1}{n} \log p(x_1, \ldots, x_n) - H| \leq \varepsilon \end{aligned} \tag{$\dagger$}$$

Then both the AEP and ($*$) say that

$$\mathbb{P}\big((X_1, \ldots, X_n) \text{ satisfies } (\dagger)\big) \to 1 \text{ as } n \to \infty.$$

□

**Theorem 3.5** (Shannon's First Coding Theorem)**.** If a source $X_1, X_2, \ldots$ satisfies the AEP with constant $H$ then it has information rate $H$.

*Proof.* Let $\varepsilon > 0$ and $T_n \subset \Sigma^n$ be typical sets. Then for all $(x_1, \ldots, x_n) \in T_n$

$$\begin{aligned} p(x_1, \ldots, x_n) &\geq 2^{-n(H+\varepsilon)} \\ \therefore \quad 1 &\geq |T_n| 2^{-n(H+\varepsilon)} \\ \therefore \quad \frac{\log |T_n|}{n} &\leq n(H + \varepsilon) \end{aligned}$$

Taking $A_n = T_n$ shows that the source is reliably encodeable at rate $H + \varepsilon$.

Conversely, if $H = 0$ we are done, otherwise pick $0 < \varepsilon < \frac{H}{2}$. We suppose for a contradiction that the source is reliably encodable at rate $H - 2\varepsilon$, say with sets $A_n \subset \Sigma^n$. Let $T_n \subset \Sigma^n$ be typical sets. Then for all $(x_1, \ldots, x_n) \in T_n$,

$$\begin{aligned} p(x_1, \ldots, x_n) &\leq 2^{-n(H-\varepsilon)} \\ \therefore \quad \mathbb{P}(A_n \cap T_n) &\leq 2^{-n(H-\varepsilon)} |A_n| \\ \therefore \quad \frac{\log \mathbb{P}(A_n \cap T_n)}{n} &\leq -(H - \varepsilon) + \frac{\log |A_n|}{n} \xrightarrow{n \to \infty} -(H - \varepsilon) + (H - 2\varepsilon) = -\varepsilon \\ \therefore \quad \log \mathbb{P}(A_n \cap T_n) &\to -\infty \text{ as } n \to \infty \\ \therefore \quad \mathbb{P}(A_n \cap T_n) &\to 0 \text{ as } n \to \infty \end{aligned}$$

But $\mathbb{P}(T_n) \leq \mathbb{P}(A_n \cap T_n) + \mathbb{P}(\Sigma^n \setminus A_n) \to 0$ as $n \to \infty$, contradicting that the $T_n$ are typical. Therefore, we cannot reliably encode at rate $H - 2\varepsilon$. Thus the information rate is $H$.

□

## Entropy as an Expectation

**Note 6.** For the entropy $H$, we have $H(X) = \mathbb{E}\big(-\log p(X)\big)$, e.g. if $X, Y$ independent,

$$\begin{aligned} p(X,Y) &= p(X)p(Y) \\ \therefore \quad -\log p(X,Y) &= -\log p(X)p(Y) \\ \therefore \quad H(X,Y) &= H(X) + H(Y), \end{aligned}$$

recovering Lemma 1.9.

**Corollary 3.6.** A Bernoulli source $X_1, X_2, \dots$ has information rate $H = H(X_1)$.

*Proof.* We have

$$\begin{aligned} p(X_1, \dots, X_n) &= p(X_1) \cdots p(X_n) \\ -\frac{1}{n} \log p(X_1, \dots, X_n) &= -\frac{1}{n} \sum_{i=1}^{n} \log p(X_i) \xrightarrow{\mathbb{P}} H(X_1) \end{aligned}$$

by the WLLN, using that $X_1, \dots$ are independent identically distributed random variables and hence so are $-\log p(X_1), \dots$.

Carefully writing out the definition of convergence in probability shows that the AEP holds with constant $H(X_1)$. (This is left as an exercise.) We conclude using Shannon's First Coding Theorem. $\square$

**Remark.** The AEP is useful for noiseless coding. We can

- encode the typical sequences using a block code;
- encode the atypical sequences arbitrarily.

**Remark.** Many sources, which are not necessarily Bernoulli, satisfy the AEP. Under suitable hypotheses, the sequence $\frac{1}{n} H(X_1, \dots, X_n)$ is decreasing and the AEP is satisfied with constant

$$H = \lim_{n \to \infty} \frac{1}{n} H(X_1, \dots, X_n).$$

**Note 7.** For a Bernoulli source $H(X_1, \dots, X_n) = nH(X_1)$.

**Example.** If our source is English text with $\Sigma = \{A, B, \dots, Z, \text{␣}\}$ then experiments show

$$\begin{aligned} H(X_1) &\approx 4.03 \\ \tfrac{1}{2} H(X_1, X_2) &\approx 3.32 \\ \tfrac{1}{3} H(X_1, X_2, X_3) &\approx 3.10 \end{aligned}$$

It is generally believed that English has entropy a bit bigger than 1, so about 75% redundancy as $1 - \frac{H}{\log|\Sigma|} \approx 1 - \frac{1}{4} = \frac{3}{4}$).

**Definition.** Consider a communication channel, with input alphabet $\Sigma_1$, output alphabet $\Sigma_2$. A *code* of *length* $n$ is a subset $C \subset \Sigma_1^n$. The *error rate* is

$$\hat{e}(C) = \max_{c \in C} \mathbb{P}(\text{error} \mid c \text{ sent}).$$

The *information rate* is

$$\rho(C) = \frac{\log|C|}{n}.$$

**Definition.** A channel can *transmit reliably at rate $R$* if there exist codes $C_1$, $C_2$, ... with $C_n$ of length $n$ such that

(i) $\lim_{n\to\infty} \rho(C_n) = R$;
(ii) $\lim_{n\to\infty} \hat{e}(C_n) = 0$.

**Definition.** The *capacity* of the channel is the supremum of all reliable transmission rates.

Suppose we are given a source

- information rate $r$ bits per symbol
- emits symbols at $s$ symbols per second

and a channel

- capacity $R$ bits per transmission
- transmits symbols at $S$ transmissions per second

Usually, mathematicians take $S = s = 1$. If $rs \leq RS$ then you can encode and transmit reliably; if $rs > RS$ then you cannot.

**Proposition 3.7.** A BSC with error probability $p < \frac{1}{4}$ has non-zero capacity.

*Proof.* The idea is to use the GSV bound. Pick $\delta$ with $2p < \delta < \frac{1}{2}$. We will show reliable transmission at rate $R = 1 - H(\delta) > 0$. Let $C_n$ be a code of length $n$ and minimum distance $\lfloor n\delta \rfloor$ of maximal size. Then

$$\begin{aligned} |C_n| = A(n, \lfloor n\delta \rfloor) &\geq 2^{n(1-H(\delta))} \quad \text{by Proposition 2.9 (ii)} \\ &= 2^{nR} \end{aligned}$$

Using minimum distance decoding,

$$\hat{e}(C_n) \leq \mathbb{P}(\text{BSC makes more than } \tfrac{n\delta - 1}{2} \text{ errors})$$

Pick $\varepsilon > 0$ with $p + \varepsilon < \frac{\delta}{2}$. For $n$ sufficiently large,

$$\begin{aligned} \frac{n\delta - 1}{2} &> n(p + \varepsilon) \\ \therefore \quad \hat{e}(C_n) &\leq \mathbb{P}(\text{BSC makes more than } n(p+\varepsilon) \text{ errors}) \\ &\to 0 \text{ as } n \to \infty \end{aligned}$$

by the next lemma. $\square$

**Lemma 3.8.** Let $\varepsilon > 0$. A BSC with error probability $p$ is used to transmit $n$ digits. Then

$$\lim_{n\to\infty} \mathbb{P}(\text{BSC makes at least } n(p+\varepsilon) \text{ errors}) = 0.$$

*Proof.* Define random variables

$$U_i = \begin{cases} 1 & \text{if the } i^{th} \text{ digit is mistransmitted} \\ 0 & \text{otherwise} \end{cases}$$

We have $U_1, U_2, \dots$ are i.i.d. and

$$\mathbb{P}(U_i = 1) = p$$
$$\mathbb{P}(U_i = 0) = 1 - p$$

and so $\mathbb{E}(U_i) = p$. Therefore,

$$\mathbb{P}(\text{BSC makes more than } n(p+\varepsilon) \text{ errors}) \leq \mathbb{P}\left(|\tfrac{1}{n}\sum_{i=0}^{n} U_i - p| \geq \varepsilon\right) \to 0$$

as $n \to \infty$ by the WLLN. □

## Conditional Entropy

Let $X, Y$ be random variables taking values in alphabets $\Sigma_1, \Sigma_2$.

**Definition.** We define

$$H(X \mid Y = y) = -\sum_{x \in \Sigma_1} \mathbb{P}(X = x \mid Y = y) \log \mathbb{P}(X = x \mid Y = y)$$
$$H(X \mid Y) = \sum_{y \in \Sigma_2} \mathbb{P}(Y = y) H(X \mid Y = y)$$

**Note 8.** Note that $H(X \mid Y) \geq 0$.

**Lemma 3.9.**

$$H(X, Y) = H(X \mid Y) + H(Y).$$

*Proof.*

$$\begin{aligned}
H(X \mid Y) &= -\sum_{x \in \Sigma_1} \sum_{y \in \Sigma_2} \mathbb{P}(X = x \mid Y = y)\, \mathbb{P}(Y = y) \log \mathbb{P}(X = x \mid Y = y) \\
&= -\sum_{x \in \Sigma_1} \sum_{y \in \Sigma_2} \mathbb{P}(X = y, Y = y) \log \left( \frac{\mathbb{P}(X = x, Y = y)}{\mathbb{P}(Y = y)} \right) \\
&= -\sum_{(x,y) \in \Sigma_1 \times \Sigma_2} \mathbb{P}(X = x, Y = y) \log \mathbb{P}(X = x, Y = y) \\
&\quad + \sum_{y \in \Sigma_2} \underbrace{\sum_{x \in \Sigma_1} \mathbb{P}(X = x, Y = y)}_{\mathbb{P}(Y=y)} \log \mathbb{P}(Y = y) \\
&= H(X, Y) - H(Y)
\end{aligned}$$

□

**Corollary 3.10.** $H(X \mid Y) \leq H(X)$ with equality if and only if $X, Y$ are independent.

*Proof.* Combine Lemma 1.9 and Lemma 3.9. □

Replacing $X, Y$ by random vectors $X_1, \dots, X_r$ and $Y_1, \dots, Y_s$, we similarly define $H(X_1, \dots, X_r \mid Y_1, \dots, Y_s)$.

**Note 9.** $H(X, Y \mid Z)$ denotes the entropy of $X$ and $Y$ given $Z$, *not* the entropy of $X$ and $Y \mid Z$.

**Lemma 3.11.** Let $X, Y, Z$ be random variables. Then

$$H(X \mid Y) \leq H(X \mid Y, Z) + H(Z).$$

*Proof.* We use Lemma 3.9 to give

$$H(X,Y,Z) = H(Z \mid X,Y) + \underbrace{H(X \mid Y) + H(Y)}_{H(X,Y)}$$
$$H(X,Y,Z) = H(X \mid Y,Z) + \underbrace{H(Z \mid Y) + H(Y)}_{H(Y,Z)}$$

Since $H(Z \mid X, Y) \geq 0$, we get

$$\begin{aligned} H(X \mid Y) &\leq H(X \mid Y, Z) + H(Z \mid Y) \\ &\leq H(X \mid Y, Z) + H(Z) \end{aligned}$$

□

**Lemma 3.12** (Fano's Inequality)**.** Let $X, Y$ be random variables taking values in $\Sigma$, $|\Sigma| = m$ say. Let $p = \mathbb{P}(X \neq Y)$. Then

$$H(X \mid Y) \leq H(p) + p \log(m-1)$$

*Proof.* Let

$$Z = \begin{cases} 0 & \text{if } X = Y \\ 1 & \text{if } X \neq Y \end{cases}$$

Then $\mathbb{P}(Z = 0) = 1 - p$, $\mathbb{P}(Z = 1) = p$ and so $H(Z) = H(p)$. Now by Lemma 3.11,

$$H(X \mid Y) \leq H(p) + H(X \mid Y, Z) \tag{$*$}$$

Since we must have $X = y$,

$$H(X \mid Y = y, Z = 0) = 0.$$

There are just $m - 1$ possibilities for $X$ and so

$$H(X \mid Y = y, Z = 1) \leq \log(m-1).$$

Therefore,

$$\begin{aligned} H(X \mid Y, Z) &= \sum_{y,z} \mathbb{P}(Y = y, Z = z) H(X \mid Y = y, Z = z) \\ &\leq \sum_{y} \mathbb{P}(Y = y, Z = 1) \log(m-1) \\ &= \mathbb{P}(Z = 1) \log(m-1) \\ &= p \log(m-1) \end{aligned}$$

Now by ($*$),

$$H(X \mid Y) \leq H(p) + p \log(m-1).$$

□

**Definition.** Let $X, Y$ be random variables. The *mutual information* is

$$I(X;Y) = H(X) - H(X \mid Y).$$

By Lemma 1.9 and Lemma 3.9,

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \geq 0,$$

with equality if and only if $X, Y$ are independent. Note the symmetry $I(X;Y) = I(Y;X)$.

Consider a DMC with input alphabet $\Sigma_1$, $|\Sigma_1| = m$, and output alphabet $\Sigma_2$. Let $X$ be a random variable taking values in $\Sigma_1$ used as input to the channel. Let $Y$ be the random variable output, depending on $X$ and the channel matrix.

**Definition.** The *information capacity* is $\max_X I(X;Y)$.

**Remark.** (i) We maximise over all probability distributions $p_1, \dots, p_m$.

(ii) The maximum is attained since we have a continuous function $I$ on a compact set

$$\left\{(p_1, \dots, p_m) \in \mathbb{R}^m : \forall i \; p_i \geq 0; \sum p_i = 1\right\}.$$

(iii) The information capacity depends only on the channel matrix.

**Theorem 3.13** (Shannon's Second Coding Theorem)**.** For a DMC, the capacity equals the information capacity.

**Note 10.** We will prove $\leq$ in general and $\geq$ for a BSC only.

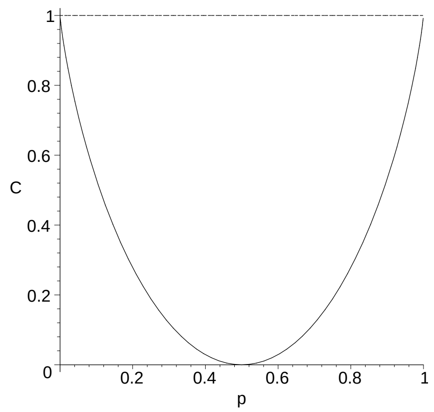
**Example.** Consider a BSC with error probability $p$, input $X$ and output $Y$.

$$\begin{aligned} \mathbb{P}(X = 0) &= \alpha & \qquad \mathbb{P}(Y = 0) &= \alpha(1-p) + (1-\alpha)p \\ \mathbb{P}(X = 1) &= 1 - \alpha & \qquad \mathbb{P}(Y = 1) &= (1-\alpha)(1-p) + \alpha p \end{aligned}$$

Then

$$\begin{aligned} C = \max_\alpha I(X;Y) &= \max_\alpha \big(H(Y) - H(Y \mid X)\big) \\ &= \max_\alpha \big(H(\alpha(1-p) + (1-\alpha)p) - H(p)\big) \\ &= 1 - H(p) \end{aligned}$$

where the maximum is attained for $\alpha = \frac{1}{2}$. Hence $C = 1 + p \log p + (1-p)\log(1-p)$ and this has the following graph.



**Note 11.** We can choose either $H(Y) - H(Y \mid X)$ or vice versa. Often one is easier.

**Example.** Consider a binary erasure channel with erasure probability $p$, input $X$ and output $Y$.

$$\mathbb{P}(X = 0) = \alpha \qquad \mathbb{P}(Y = 0) = \alpha(1-p)$$
$$\mathbb{P}(X = 1) = 1 - \alpha \qquad \mathbb{P}(Y = \star) = p$$
$$\mathbb{P}(Y = 1) = (1-\alpha)(1-p)$$

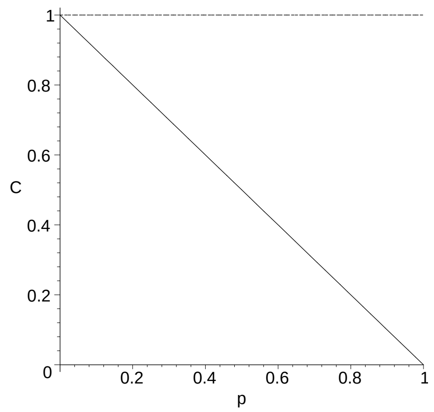
Then

$$\begin{aligned} H(X \mid Y = 0) &= 0 \\ H(X \mid Y = \star) &= H(\alpha) \\ H(X \mid Y = 1) &= 0 \end{aligned}$$

and hence $H(X \mid Y) = pH(\alpha)$. Therefore,

$$\begin{aligned} C = \max_\alpha I(X;Y) &= \max_\alpha \big(H(X) - H(X \mid Y)\big) \\ &= \max_\alpha H(\alpha) - pH(\alpha) \\ &= 1 - p \end{aligned}$$

where the maximum is attained for $\alpha = \frac{1}{2}$. This has the following graph.




**Lemma 3.14.** The $n$th extension of a DMC with information capacity $C$ has information capacity $nC$.

*Proof.* The input $X_1, \dots, X_n$ determines the ouput $Y_1, \dots, Y_n$. Since the channel is memoryless,

$$\begin{aligned} H(Y_1, \dots, Y_n \mid X_1, \dots, X_n) &= \sum_{i=1}^n H(Y_i \mid X_1, \dots, X_n) \\ &= \sum_{i=1}^n H(Y_i \mid X_i) \\ I(X_1, \dots, X_n; Y_1, \dots, Y_n) &= H(Y_1, \dots, Y_n) - H(Y_1, \dots, Y_n \mid X_1, \dots, X_n) \\ &= H(Y_1, \dots, Y_n) - \sum_{i=1}^n H(Y_i \mid X_i) \end{aligned}$$

$$\begin{aligned}
&\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i \mid X_i) \\
&= \sum_{i=1} I(X_i, Y_i) \\
&\leq nC
\end{aligned}$$

We now need to find $X_1, \dots, X_n$ giving equality to complete the proof. Equality is attained by taking $X_1, \dots, X_n$ independent, each with the same distribution such that $I(X_i; Y_i) = C$. Indeed, if $X_1, \dots, X_n$ are independent then $Y_1, \dots, Y_n$ are independent, so

$$H(Y_1, \dots, Y_n) = \sum_{i=1}^{n} H(Y_i)$$

and we have equality. Therefore,

$$\max_{X_1,\dots,X_n} I(X_1, \dots, X_n; Y_1, \dots, Y_n) = nC.$$

□

**Proposition 3.15.** For a DMC, the capacity is at most the information capacity.

*Proof.* Let $C$ be the information capacity. Suppose reliable transmission is possible at some rate $R > C$, i.e. there exist $C_1, C_2, \dots$ with $C_n$ of length $n$ such that

$$\begin{aligned}
\lim_{n\to\infty} \rho(C_n) &= R \\
\lim_{n\to\infty} \hat{e}(C_n) &= 0
\end{aligned}$$

Recall

$$\hat{e}(C_n) = \max_{c \in C_n} \mathbb{P}(\text{error} \mid c \text{ sent}).$$

Now consider the *average error rate*

$$e(C_n) = \frac{1}{|C_n|} \sum_{c \in C_n} \mathbb{P}(\text{error} \mid c \text{ sent}).$$

Clearly $e(C_n) \leq \hat{e}(C_n)$ and so $e(C_n) \to 0$ as $n \to \infty$.

Let $X$ be a random variable equidistributed in $C_n$. We transmit $X$ and decode to obtain $Y$. So $e(C_n) = \mathbb{P}(X \neq Y)$. Then

$$\begin{aligned}
H(X) = \log|C_n| &= \log \lfloor 2^{nR} \rfloor \\
&\geq nR - 1
\end{aligned}$$

for $n$ sufficiently large. Thus by Fano's inequality 3.12,

$$\begin{aligned}
H(X \mid Y) &\leq \underbrace{1}_{H(p)\leq 1} + e(C_n) \log(|C_n| - 1) \\
&\leq 1 + e(C_n) n\rho(C_n)
\end{aligned}$$

since $|C_n| = \lfloor 2^{nR} \rfloor$. Now by Lemma 3.14,

$$nC \geq I(X; Y)$$

$$
\begin{aligned}
&= H(X) - H(X \mid Y) \\
&\geq \log|C_n| - (1 + e(C_n)n\rho(C_n)) \\
&= n\rho(C_n) + e(C_n)n\rho(C_n) - 1 \\
\therefore \quad e(C_n)n\rho(C_n) &\geq n(\rho(C_n) - C) - 1 \\
e(C_n) &\geq \frac{\rho(C_n) - C}{\rho(C_n)} - \frac{1}{n\rho(C_n)} \to \frac{R - C}{R} \text{ as } n \to \infty
\end{aligned}
$$

Since $R > C$, this contradicts $e(C_n) \to 0$ as $n \to \infty$. This shows that we cannot transmit reliably at any rate $R > C$, hence the capacity is at most $C$. □

To complete the proof of Shannon's Second Coding Theorem for a BSC with error probability $p$, we must show that the capacity is at most $1 - H(p)$.

**Proposition 3.16.** Consider a BSC with error probability $p$. Let $R < 1 - H(p)$. Then there exists codes $C_1, C_2, \ldots$ with $C_n$ of length $n$ and such that

$$
\begin{aligned}
\lim_{n\to\infty} \rho(C_n) &= R \\
\lim_{n\to\infty} e(C_n) &= 0
\end{aligned}
$$

**Note 12.** Note that Proposition 3.16 is concerned with with the average error rate $e$ rather than the error rate $\hat{e}$.

*Proof.* The idea of the proof is to pick codes at random. Without loss of generality, assume $p < \frac{1}{2}$. Take $\varepsilon > 0$ such that

$$
\begin{aligned}
p + \varepsilon &< \frac{1}{2} \\
R &< 1 - H(p + \varepsilon)
\end{aligned}
$$

Note this is possible since $H$ is continuous. Let $m = \lfloor 2^{nR} \rfloor$ and $\Gamma$ be the set of $[n, m]$-codes, so $|\Gamma| = \binom{2^n}{m}$. Let $\mathfrak{C}$ be a random variable equidistributed in $\Gamma$. Say $\mathfrak{C} = \{X_1, \ldots, X_m\}$ where the $X_i$ are random variables taking values in $\mathbb{F}_2^n$ such that

$$
\mathbb{P}(X_i = x \mid \mathfrak{C} = C) = \begin{cases} \frac{1}{m} & \text{if } x \in C \\ 0 & \text{otherwise} \end{cases}
$$

Note that

$$
\mathbb{P}(X_2 = x_2 \mid X_1 = x_1) = \begin{cases} \frac{1}{2^n - 1} & x_1 \neq x_2 \\ 0 & x_1 = x_2 \end{cases}
$$

We send $X = X_1$ through the BSC, receive $Y$ and decode to obtain $Z$. Using minimum distance decoding,

$$
\mathbb{P}(X \neq Z) = \frac{1}{|\Gamma|} \sum_{C \in \Gamma} e(C)
$$

It suffices to show that $\mathbb{P}(X \neq Z) \to 0$ as $n \to \infty$. Let $r = \lfloor n(p + \varepsilon) \rfloor$.

$$
\begin{aligned}
\mathbb{P}(X \neq Z) &\leq \mathbb{P}(B(Y, r) \cap \mathfrak{C} \neq \{X\}) \\
&= \mathbb{P}(X \notin B(Y, r)) + \mathbb{P}(B(Y, r) \cap \mathfrak{C} \supsetneq \{X\})
\end{aligned}
$$

We consider the two terms on the RHS separately.

$$\begin{aligned}\mathbb{P}(X \notin B(Y,r)) &= \mathbb{P}(\text{BSC makes more than } n(p+\varepsilon) \text{ errors})\\ &\to 0\end{aligned}$$

as $n \to \infty$, by the WLLN, see Lemma 3.8.

$$\begin{aligned}\mathbb{P}(B(Y,r) \cap \mathfrak{C} \supsetneq \{X\}) &\le \sum_{i=2}^{m} \mathbb{P}(X_i \in B(Y,r) \text{ and } X_1 \in B(Y,r))\\ &\le \sum_{i=2}^{m} \mathbb{P}(X_i \in B(Y,r) \mid X_1 \in B(Y,r))\\ &= (m-1)\frac{V(n,r)-1}{2^n-1}\\ &\le m\frac{V(n,r)}{2^n}\\ &\le 2^{nR}2^{nH(p+\varepsilon)}2^{-n}\\ &= 2^{n[R-(1-H(p+\varepsilon))]}\\ &\to 0,\end{aligned}$$

as $n \to \infty$ since $R < 1 - H(p+\varepsilon)$. We have used Proposition 2.9 to obtain the last inequality. □

**Proposition 3.17.** We can replace $e$ by $\hat{e}$ in Proposition 3.16.

*Proof.* Pick $R'$ with $R < R' < 1 - H(p)$. Proposition 3.16 constructs $C_1', C_2', \ldots$ with $C_n'$ of length $n$, size $\lfloor 2^{nR'} \rfloor$ and $e(C_n') \to 0$ as $n \to \infty$. Order the codewords of $C_n'$ by $\mathbb{P}(\text{error} \mid c \text{ sent})$ and delete the worse half of them to give $C_n$. We have

$$|C_n| = \left\lfloor \frac{|C_n'|-1}{2} \right\rfloor$$

and

$$\hat{e}(C_n) \le 2e(C_n')$$

Then $\rho(C_n) \to R$ and $\hat{e}(C_n) \to 0$ as $n \to \infty$. □

Proposition 3.17 says that we can transmit reliably at any rate $R < 1 - H(p)$, so the capacity is at least $1 - H(p)$. But by Proposition 3.15, the capacity is at most $1 - H(p)$, hence a BSC with error probability $p$ has capacity $1 - H(p)$.

**Remark.** The proof shows that good codes exist, but does not tell us how to construct them.

# Chapter 4

# Linear and Cyclic Codes

**Definition.** A code $C \subset \mathbb{F}_2^n$ is *linear* if

(i) $0 \in C$;
(ii) whenever $x, y \in C$ then $x + y \in C$.

Equivalently, $C$ is an $\mathbb{F}_2$-vector subspace of $\mathbb{F}_2^n$.

**Definition.** The *rank* of $C$ is its dimension as a $\mathbb{F}_2$-vector subspace. A linear code of length $n$, rank $k$ is an $(n, k)$*-code*. If the minimum distance is $d$, it is an $(n, k, d)$*-code*.

Let $v_1, \dots, v_k$ be a basis for $C$. Then

$$C = \left\{ \sum_{i=1}^{k} \lambda_i v_i : \lambda_1, \dots, \lambda_k \in \mathbb{F}_2 \right\},$$

so $|C| = 2^k$. So an $(n, k)$-code is an $[n, 2^k]$-code. The information rate is $\rho(C) = \frac{k}{n}$.

**Definition.** The *weight* of $x \in \mathbb{F}_2^n$ is $\omega(x) = d(x, 0)$.

**Lemma 4.1.** The minimum distance of a linear code is the minimum weight of a non-zero codeword.

*Proof.* If $x, y \in C$ then $d(x, y) = d(x + y, 0) = \omega(x + y)$. Therefore,

$$\min\{d(x, y) : x, y \in C, x \neq y\} = \min\{\omega(c) : c \in C : c \neq 0\}. \qquad \square$$

**Notation.** For $x, y \in \mathbb{F}_2^n$, let $x.y = \sum_{i=1}^n x_i y_i \in \mathbb{F}_2$. Beware that there exists $x \neq 0$ with $x.x = 0$.

**Definition.** Let $P \subset \mathbb{F}_2^n$. The *parity check code* defined by $P$ is

$$C = \{x \in \mathbb{F}_2^n : p.x = 0 \ \forall p \in P\}.$$

**Example.** (i) $P = \{111 \dots 1\}$ gives the simple parity check code.
(ii) $P = \{1010101, 0110011, 0001111\}$ gives Hamming's $[7, 16, 3]$-code.

**Lemma 4.2.** Every parity check code is linear.

*Proof.* $0 \in C$ since $p.0 = 0 \ \forall p \in P$. If $x, y \in C$ then

$$p.(x + y) = p.x + p.y = 0 \quad \forall p \in P. \qquad \square$$

**Definition.** Let $C \subset \mathbb{F}_2^n$ be a linear code. The *dual code* is

$$C^\perp = \{x \in \mathbb{F}_2^n : x.y = 0 \ \forall y \in C\}.$$

This is a parity check code, so it is linear. Beware that we can have $C \cap C^\perp \neq \{0\}$.

**Lemma 4.3.** $\operatorname{rank} C + \operatorname{rank} C^\perp = n$.

*Proof.* We can use the similar result about dual spaces ($C^\perp = \operatorname{Ann} C$) from *Linear Algebra*. An alternative proof is presented on Page 33. □

**Lemma 4.4.** Let $C$ be a linear code. Then $(C^\perp)^\perp = C$. In particular, $C$ is a parity check code.

*Proof.* Let $x \in C$. Then $x.y = 0 \ \forall y \in C^\perp$. So $x \in (C^\perp)^\perp$, hence $C \subset (C^\perp)^\perp$. By Lemma 4.3,

$$\operatorname{rank} C = n - \operatorname{rank} C^\perp = n - (n - \operatorname{rank}(C^\perp)^\perp) = \operatorname{rank}(C^\perp)^\perp.$$

So $C = (C^\perp)^\perp$. □

**Definition.** Let $C$ be an $(n, k)$-code.

(i) A *generator matrix* $G$ for $C$ is a $k \times n$ matrix with rows a basis for $C$.
(ii) A *parity check matrix* $H$ for $C$ is a generator matrix for $C^\perp$. It is an $(n-k) \times n$ matrix.

The codewords in $C$ can be views as

(i) linear combinations of rows of $G$;
(ii) linear dependence relations between the columns of $H$, i.e.

$$C = \{x \in \mathbb{F}_2^n : Hx = 0\}.$$

## Syndrome Decoding

Let $C$ be an $(n, k)$-linear code. Recall that

- $C = \{G^T y : y \in \mathbb{F}_2^k\}$ where $G$ is the generator matrix.
- $C = \{x \in \mathbb{F}_2^n : Hx = 0\}$ where $H$ is the parity check matrix.

**Definition.** The *syndrome* of $x \in \mathbb{F}_2^n$ is $Hx$.

If we receive $x = c + z$, where $c$ is the codeword and $z$ is the error pattern, then $Hx = Hc + Hz = Hz$. If $C$ is $e$-error correcting, we precompute $Hz$ for all $z$ with $\omega(z) \leq e$. On receiving $x \in \mathbb{F}_2^n$, we look for $Hx$ in our list. $Hx = Hz$, so $H(x - z) = 0$, so $c = x - z \in C$ with $d(x, c) = \omega(z) \leq e$.

**Remark.** We did this for Hamming's $(7, 4)$-code, where $e = 1$.

**Definition.** Codes $C_1, C_2 \in \mathbb{F}_2^n$ are *equivalent* if reordering each codeword of $C_1$ using the same permutation gives the codewords of $C_2$.

**Lemma 4.5.** Every $(n,k)$-linear code is equivalent to one with generator matrix $G = (I_k \mid B)$ for some $k \times (n-k)$ matrix $B$.

*Proof.* Using Gaussian elimination, i.e. row operations, we can transform $G$ into row echelon form, i.e.

$$G_{ij} = \begin{cases} 0 & \text{if } j < l(i) \\ 1 & \text{if } j = l(i) \end{cases}$$

for some $l(1) < l(2) < \cdots < l(k)$. Permuting columns replaces the code by an equivalent code. So without loss of generality we may assume $l(i) = i$ for all $1 \le i \le k$. Therefore,

$$G = \left( \begin{array}{ccc|c} 1 & & * & \\ & \ddots & & * \\ 0 & & 1 & \end{array} \right)$$

Further row operations give $G = (I_k | B)$. $\square$

**Remark.** A message $y \in \mathbb{F}_2^k$, viewed as a row vector, is encoded as $yG$. So if $G = (I_k \mid B)$ then $yG = (y \mid yB)$, where $y$ is the message and $yB$ are the check digits.

*Proof of Lemma 4.3.* Without loss of generality $C$ has generator matrix $G = (I_k \mid B)$. $G$ has $k$ linearly independent columns, so the linear map $\gamma\colon \mathbb{F}_2^n \to \mathbb{F}_2^k, x \mapsto Gx$ is surjective and $\ker(\gamma) = C^\perp$, so by the rank-nullity theorem we obtain

$$\begin{aligned} \dim \mathbb{F}_2^n &= \dim\ker(\gamma) + \dim \operatorname{Im}(\gamma) \\ n &= \operatorname{rank} C^\perp + \operatorname{rank} C. \end{aligned}$$ $\square$

**Lemma 4.6.** An $(n,k)$-linear code with generator matrix $G = (I_k \mid B)$ has parity check matrix $H = (B^T \mid I_{n-k})$.

*Proof.* Since $GH^T = (I_k \mid B)\left(\frac{B}{I_{n-k}}\right) = B + B = 0$, the rows of $H$ generate a subcode of $C^\perp$. But $\operatorname{rank} H = n-k$ since $H$ contains $I_{n-k}$, and $n - k = \operatorname{rank} C^\perp$ by Lemma 4.3, so $C^\perp$ has generator matrix $H$. $\square$

**Remark.** We usually only consider codes up to equivalence.

## Hamming Codes

**Definition.** For $d \ge 1$, let $n = 2^d - 1$. Let $H$ be the $d \times n$ matrix whose columns are the non-zero elements of $\mathbb{F}_2^d$. The *Hamming* $(n, n-d)$*-code* is the linear code with parity check matrix $H$. Note this is only defined up to equivalence.

**Example.** For $d = 3$, we have

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

**Lemma 4.7.** The minimum distance of the $(n, n-d)$ Hamming code $C$ is $d(C) = 3$. It is a perfect 1-error correcting code.

*Proof.* The codewords of $C$ are dependence relations between the columns of $H$. Any two columns of $H$ are linearly independent, so there are no non-zero codewords of weight at most 2. Hence $d(C) \geq 3$. If

$$H = \begin{pmatrix} 1 & 0 & 1 & \dots \\ 0 & 1 & 1 & \\ \vdots & & & \end{pmatrix}$$

then $1110\dots0 \in C$, hence $d(C) = 3$.

By Lemma 2.4, $C$ is 1-error correcting. But

$$\frac{2^n}{V(n,1)} = \frac{2^n}{n+1} = 2^{n-d} = |C|,$$

so $C$ is perfect. $\square$

## Reed–Muller Codes

Take a set $X$ such that $|X| = n$, $X = \{P_1, \dots, P_n\}$. There is a correspondence between $\mathcal{P}(X)$ and $\mathbb{F}_2^n$.

$$\begin{tikzcd}
\mathcal{P}(X) \arrow[r, leftrightarrow] & \{f \colon X \to \mathbb{F}_2\} \arrow[r, leftrightarrow] & \mathbb{F}_2^n \\
A \arrow[r, mapsto] & 1_A & \\
 & f \arrow[r, mapsto] & (f(P_1), \dots, f(P_n))
\end{tikzcd}$$

$$\begin{tikzcd}
\begin{matrix}\text{symmetric difference} \\ A \triangle B = (A \setminus B) \cup (B \setminus A)\end{matrix} \arrow[r, leftrightarrow, dashed] & \begin{matrix}\text{vector addition} \\ x + y = (x_1 + y_1, \dots, x_n + y_n)\end{matrix} \\
\begin{matrix}\text{intersection} \\ A \cap B\end{matrix} \arrow[r, leftrightarrow, dashed] & \begin{matrix}\text{wedge product} \\ x \wedge y = (x_1 y_1, \dots, x_n y_n)\end{matrix}
\end{tikzcd}$$

Take $X = \mathbb{F}_2^d$, so $n = |X| = 2^d$. Let $v_0 = 1_X = (1, 1, \dots, 1)$. Let $v_i = 1_{H_i}$ for $1 \leq i \leq d$ where $H_i = \{p \in X : p_i = 0\}$, called the *coordinate hyperplane*.

**Definition.** The *Reed–Muller code $RM(d, r)$ of order $r$*, $0 \leq r \leq d$, and length $2^d$ is the linear code spanned by $v_0$ and all wedge products of $r$ or fewer of the $v_i$. By convention, the empty wedge product is $v_0$.

**Example.** Let $d = 3$.

| $X$ | {000 | 001 | 010 | 011 | 100 | 101 | 110 | 111} |
|---|---|---|---|---|---|---|---|---|
| $v_0$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $v_1$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| $v_2$ | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| $v_3$ | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| $v_1 \wedge v_2$ | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| $v_2 \wedge v_3$ | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $v_1 \wedge v_3$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $v_1 \wedge v_2 \wedge v_3$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

This gives the following codes.

- $RM(3,0)$ is spanned by $v_0$, a repetion code of length 8.
- $RM(3,1)$ is spanned by $v_0, v_1, v_2, v_3$, a parity check extension of Hamming's $(7,4)$-code.
- $RM(3,2)$ is an $(8,7)$-code, in fact it is the simple parity check code.
- $RM(3,3)$ is $\mathbb{F}_2^8$, the trivial code.

**Theorem 4.8.** (i) The vectors $v_{i_1} \wedge \cdots \wedge v_{i_s}$ for $1 \leq i_1 < i_2 < \cdots < i_s \leq d$ and $0 \leq s \leq d$ are a basis for $\mathbb{F}_2^n$.
(ii) $\operatorname{rank} RM(d,r) = \sum_{s=0}^{r} \binom{d}{s}$.

*Proof.* (i) We have listed $\sum_{i=0}^{d} \binom{d}{s} = (1+1)^d = 2^d = n$ vectors, so it suffices to check spanning, i.e. check $RM(d,d) = \mathbb{F}_2^n$. Let $p \in X$ and

$$y_i = \begin{cases} v_i & \text{if } p_i = 0 \\ v_0 + v_i & \text{if } p_1 = 1 \end{cases}$$

Then $1_{\{p\}} = y_1 \wedge \cdots \wedge y_d$. Expand this using the distributive law to show $1_{\{p\}} \in RM(d,d)$. But $1_{\{p\}}$ for $p \in X$ span $\mathbb{F}_2^n$, so the given vectors form a basis.
(ii) $RM(d,r)$ is spanned by the vectors $v_{i_1} \wedge \cdots \wedge v_{i_s}$ for $1 \leq i_1 < \cdots < i_s \leq d$ with $0 \leq s \leq r$. These vectors are linearly independent by (i), so a basis. Therefore, $\operatorname{rank} RM(d,r) = \sum_{s=0}^{r} \binom{d}{s}$. □

**Definition.** Let $C_1, C_2$ be linear codes of length $n$ with $C_2 \subset C_1$. The *bar product* is $C_1 \mid C_2 = \{(x \mid x+y) : x \in C_1, y \in C_2\}$. It is a linear code of length $2n$.

**Lemma 4.9.** (i) $\operatorname{rank}(C_1 \mid C_2) = \operatorname{rank} C_1 + \operatorname{rank} C_2$;
(ii) $d(C_1 \mid C_2) = \min\{2d(C_1), d(C_2)\}$.

*Proof.* (i) $C_1$ has basis $x_1, \ldots, x_k$, $C_2$ has basis $y_1, \ldots, y_l$. $C_1 \mid C_2$ has basis $\{(x_i \mid x_i) : 1 \leq i \leq k\} \cup \{(0 \mid y_i) : 1 \leq i \leq l\}$. Therefore,

$$\operatorname{rank}(C_1 \mid C_2) = k + l = \operatorname{rank} C_1 + \operatorname{rank} C_2.$$

(ii) Let $0 \neq (x \mid x+y) \in C_1 \mid C_2$. If $y \neq 0$ then $\omega(x \mid x+y) \geq \omega(y) \geq d(C_2)$. If $y = 0$ then $\omega(x \mid x+y) = 2\omega(x) \geq 2d(C_1)$. Therefore,

$$d(C_1 \mid C_2) \geq \min\{2d(C_1), d(C_2)\}.$$

There exists $x \in C_1$ with $\omega(x) = d(C_1)$. Then $d(C_1 \mid C_2) \leq \omega(x \mid x) = 2d(C_1)$. There exists $y \in C_2$ with $\omega(y) = d(C_2)$. Then $d(C_1 \mid C_2) \leq \omega(0 \mid y) = d(C_2)$. Therefore,

$$d(C_1 \mid C_2) \leq \min\{2d(C_1), d(C_2)\}.$$ □

**Theorem 4.10.** (i) $RM(d,r) = RM(d-1,r) \mid RM(d-1,r-1)$.
(ii) $RM(d,r)$ has minimum distance $2^{d-r}$.

*Proof.* (i) Note $RM(d-1,r-1) \subset RM(d-1,r)$, so the bar product is defined. Order the elements of $X = \mathbb{F}_2^d$ such that

$$v_d = (00\ldots0 \mid 11\ldots1)$$

$$v_i = (v_i' \mid v_i') \quad \text{for } 1 \le i \le d-1.$$

If $z \in RM(d, r)$, then $z$ is a sum of wedge products of $v_1, \dots, v_d$. So $z = x + y \wedge v_d$ for $x, y$ sums of wedge products of $v_1, \dots, v_{d-1}$. Then

$$\begin{aligned} x &= (x' \mid x') \quad \text{for some } x' \in RM(d-1, r) \\ y &= (y' \mid y') \quad \text{for some } y' \in RM(d-1, r-1) \end{aligned}$$

Then

$$\begin{aligned} z &= x + y \wedge v_d \\ &= (x' \mid x') + (y' \mid y') \wedge (00\dots0 \mid 11\dots1) \\ &= (x' \mid x' + y') \end{aligned}$$

So $z \in RM(d-1, r) \mid RM(d-1, r-1)$.

(ii) If $r = 0$ then $RM(d, 0)$ is a repetition code of length $n = 2^d$. This has minimum distance $2^{d-0}$. If $r = d$ then $RM(d, d) = \mathbb{F}_2^n$ with minimum distance $1 = 2^{d-d}$. We prove the case $0 < r < d$ by induction on $d$. Recall

$$RM(d, r) = RM(d-1, r) \mid RM(d-1, r-1).$$

The minimum distance of $RM(d-1, r)$ is $2^{d-1-r}$ and of $RM(d-1, r-1)$ is $2^{d-r}$. By Lemma 4.9, the minimum distance of $RM(d, r)$ is

$$\min\{2(2^{d-1-r}), 2^{d-r}\} = 2^{d-r}. \qquad \square$$

## GRM Revision: Polynomial Rings and Ideals

**Definition.** (i) A *ring* $R$ is a set with operations $+$ and $\times$, satisfying certain axioms (familiar as properties of $\mathbb{Z}$).

(ii) A *field* is a ring where every non-zero element has a multiplicative inverse, e.g. $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$, $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ for $p$ prime.

Every field is either an extension of $\mathbb{F}_p$ (with characteristic $p$) or an extension of $\mathbb{Q}$ (with characteristic 0).

**Definition.** Let $R$ be a ring. The *polynomial ring* with coefficients in $R$ is

$$R[X] = \left\{ \sum_{i=0}^{n} a_i X^i : a_0, \dots, a_n \in \mathbb{R}, n \in \mathbb{N} \right\}$$

with the usual operations.

**Remark.** By definition, $\sum_{i=0}^n a_i X^i = 0$ if and only $a_i = 0$ for all $i$. Thus $f(X) = X^2 + X \in \mathbb{F}_2[X]$ is non-zero, yet $f(a) = 0$ for all $a \in \mathbb{F}_2$.

Let $F$ be any field. The rings $\mathbb{Z}$ and $F[X]$ both have a division algorithm: if $a, b \in \mathbb{Z}$, $b \neq 0$ then there exist $q, r \in \mathbb{Z}$ such that $a = qb + r$ and $0 \le r < |b|$. If $f, g \in F[X]$, $g \neq 0$ then there exist $q, r \in F[X]$ such that $f = qg + r$ with $\deg(r) < \deg(g)$.

**Definition.** An *ideal* $I \subset R$ is a subgroup under addition such that

$$r \in R, x \in I \implies rx \in I.$$

**Definition.** The *principal ideal* generated by $x \in R$ is

$$(x) = Rx = xR = \{rx : r \in R\}.$$

By the division algorithm, every ideal in $\mathbb{Z}$ or $F[X]$ is principal, generated by an element of least absolute value respectively least degree. The generator of a principal ideal is unique up to multiplication by a unit, i.e. an element with multiplicative inverse. $\mathbb{Z}$ has units $\{\pm 1\}$, $F[X]$ has units $F \setminus \{0\}$, i.e. non-zero constants.

**Fact.** Every non-zero element of $\mathbb{Z}$ or $F[X]$ can be factored into irreducibles, uniquely up to order and multiplication by units.

If $I \subset R$ is an ideal then the set of cosets $R/I = \{x + I : x \in R\}$ is a ring, called the *quotient ring*, under the natural choice of $+$ and $\times$. In practice, we identify $\mathbb{Z}/n\mathbb{Z}$ and $\{0, 1, \dots, n-1\}$ and agree to reduce modulo $n$ after each $+$ and $\times$. Similarly,

$$F[X]/(f(X)) = \left\{ \sum_{i=0}^{n-1} a_i X^i : a_0, \dots, a_{n-1} \in F \right\} = F^n$$

where $n = \deg f$, reducing after each multiplication using the division algorithm.

# Cyclic Codes

**Definition.** A linear code $C \subset \mathbb{F}_2^n$ is *cyclic* if

$$(a_0, a_1, \dots, a_{n-1}) \in C \implies (a_{n-1}, a_0, a_1, \dots, a_{n-2}) \in C.$$

We identify

$$\mathbb{F}_2[X]/(X^n - 1) \longleftrightarrow \{f \in \mathbb{F}_2[X] : \deg f < n\} \longleftrightarrow \mathbb{F}_2^n$$

$$a_0 + a_1 X + \cdots a_{n-1} X^{n-1} \longleftrightarrow (a_0, a_1, \dots, a_{n-1})$$

**Lemma 4.11.** A code $C \subset \mathbb{F}_2[X]/(X^n - 1)$ is cyclic if and only if

(i) $0 \in C$

(ii) $f, g \in C \implies f + g \in C$

(iii) $f \in \mathbb{F}_2[X], g \in C \implies fg \in C$

Equivalently, $C$ is an ideal in $\mathbb{F}_2[X]/(X^n - 1)$.

*Proof.* If $g(X) \equiv a_0 + a_1 X + \cdots + a_{n-1} X^{n-1} \pmod{X^n - 1}$, then $Xg(X) \equiv a_{n-1} + a_0 X + \cdots + a_{n-2} X^{n-1} \pmod{X^n - 1}$. So $C$ is cyclic if and only if

(i) $0 \in C$;

(ii) $f, g \in C \implies f + g \in C$;

(iii)' $g(X) \in C \implies Xg(X) \in C$.

Note (iii)' is the case $f(X) = X$ of (iii). In general, $f(X) = \sum a_i X^i$, so

$$f(X)g(X) = \sum a_i \underbrace{X^i g(X)}_{\in C \text{ by (iii)}} \in C$$

by (ii). □

## Basic Problem

Our basic problem is to find all cyclic codes of length $n$. The following diagram outlines the solution.

$$\begin{tikzcd}
\boxed{\begin{array}{c}\text{cyclic codes}\\ \text{of length } n\end{array}} \arrow[r, leftrightarrow] & \boxed{\begin{array}{c}\text{ideals in}\\ \mathbb{F}_2[X]/(X^n-1)\end{array}} \arrow[r, leftrightarrow] & \boxed{\begin{array}{c}\text{ideals in } \mathbb{F}_2[X]\\ \text{containing } X^n-1\end{array}} \arrow[d, leftrightarrow, "\mathbb{F}_2[X] \text{ a PID}"] \\
& & \boxed{\begin{array}{c}\text{polynomials } g(X) \in \mathbb{F}_2[X]\\ \text{dividing } X^n-1\end{array}}
\end{tikzcd}$$

**Theorem 4.12.** Let $C \subset \mathbb{F}_2[X]/(X^n-1)$ be a cyclic code. Then there exists a unique $g(X) \in \mathbb{F}_2[X]$ such that

(i) $C = \{f(X)g(X) \pmod{X^n-1} : f(X) \in \mathbb{F}_2[X]\}$;
(ii) $g(X) \mid X^n-1$.

In particular, $p(X) \in \mathbb{F}_2[X]$ represents a codeword if and only if $g(X) \mid p(X)$. We say $g(X)$ is the *generator polynomial* of $C$.

*Proof.* Let $g(X) \in \mathbb{F}_2[X]$ be of least degree representing a non-zero codeword. Note $\deg g < n$. Since $C$ is cyclic, we have $\supset$ in (i).

Let $p(X) \in \mathbb{F}_2[X]$ represent a codeword. By the division algorithm, $p(X) = q(X)g(X) + r(X)$ for some $q, r \in \mathbb{F}_2[X]$ with $\deg r < \deg g$. So $r(X) = p(X) - q(X)g(X) \in C$, contradicting the choice of $g(X)$ unless $r(X)$ is a multiple of $X^n - 1$, hence $r(X) = 0$ as $\deg r < \deg g < n$; i.e. $g(X) \mid p(X)$. This shows $\subset$ in (i).

Taking $p(X) = X^n - 1$ gives (ii).

Uniqueness. Suppose $g_1(X), g_2(X)$ both satisfy (i) and (ii). Then $g_1(X) \mid g_2(X)$ and $g_2(X) \mid g_1(X)$, so $g_1(X) = ug_2(X)$ for some unit $u$. But units in $\mathbb{F}_2[X]$ are $\mathbb{F}_2 \backslash \{0\} = \{1\}$, so $g_1(X) = g_2(X)$. $\square$

**Lemma 4.13.** Let $C$ be a cyclic code of length $n$ with generator polynomial $g(X) = a_0 + a_1X + \cdots + a_kX^k$, $a_k \neq 0$. Then $C$ has basis $g(X), Xg(X), \ldots, X^{n-k-1}g(X)$. In particular, $C$ has rank $n-k$.

*Proof.* (i) Linear independence. Suppose $f(X)g(X) \equiv 0 \pmod{X^n-1}$ for some $f(X) \in \mathbb{F}_2[X]$ with $\deg f < n-k$. Then $\deg fg < n$, so $f(X)g(X) = 0$, hence $f(X) = 0$, i.e. every dependence relation is trivial.

(ii) Spanning. Let $p(X) \in \mathbb{F}_2[X]$ represent a codeword. Without loss of generality $\deg p < n$. Since $g(X)$ is the generator polynomial, $g(X) \mid p(X)$, i.e. $p(X) = f(X)g(X)$ for some $f(X) \in \mathbb{F}_2[X]$. $\deg f = \deg p - \deg g < n-k$, so $p(X)$ belongs to the span of $g(X), Xg(X), \ldots, X^{n-k-1}g(X)$. $\square$

**Corollary 4.14.** The $n \times (n-k)$ generator matrix is

$$G = \begin{pmatrix}
a_0 & a_1 & \ldots & a_k & & & & 0 \\
& a_0 & a_1 & \ldots & a_k & & & \\
& & a_0 & a_1 & \ldots & a_k & & \\
& & & \ddots & & & \ddots & \\
0 & & & & a_0 & a_1 & \ldots & a_k
\end{pmatrix}$$

**Definition.** The *parity check polynomial* $h(X) \in \mathbb{F}_2[X]$ is defined by $X^n - 1 = g(X)h(X)$.

**Note 13.** If $h(X) = b_0 + b_1 X + \cdots + b_{n-k}X^{n-k}$, then the $n \times k$ parity check matrix is

$$H = \begin{pmatrix} b_{n-k} & \dots & b_1 & b_0 & & & & 0 \\ & b_{n-k} & \dots & b_1 & b_0 & & & \\ & & b_{n-k} & \dots & b_1 & b_0 & & \\ & & & \ddots & & & \ddots & \\ 0 & & & & b_{n-k} & \dots & b_1 & b_0 \end{pmatrix}$$

Indeed, the dot product of the $i$th row of $G$ and the $j$th row of $H$ is the coefficient of $X^{(n-k-i)+j}$ in $g(X)h(X)$. But $1 \le i \le n-k$ and $1 \le j \le k$, so $0 < (n-k-i)+j < n$. These coefficients of $g(X)h(X) = X^n - 1$ are zero, hence the rows of $G$ and $H$ are orthogonal. Also $\operatorname{rank} H = k = \operatorname{rank} C^\perp$, so $H$ is a parity check matrix.

**Remark.** The check polynomial is the "reverse" of the generator polynomial for the dual code.

**Lemma 4.15.** If $n$ is odd then $X^n - 1 = f_1(X) \dots f_t(X)$ with $f_1(X), \dots, f_t(X)$ distinct irreducibles in $\mathbb{F}_2[X]$. (Note this is false for $n$ even, e.g. $X^2 - 1 = (X-1)^2$ in $\mathbb{F}_2[X]$.) In particular, there are $2^t$ cyclic codes of length $n$.

*Proof.* Suppose $X^n - 1$ has a repeated factor. Then there exists a field extension $K/\mathbb{F}_2$ such that $X^n - 1 = (X-a)^2 g(X)$ for some $a \in K$ and some $g(X) \in K[X]$. Taking formal derivatives, $nX^{n-1} = 2(X-a)g(X) + (X-a)^2 g'(X)$ so $na^{n-1} = 0$, so $a = 0$ since $n$ is odd, hence $0 = a^n = 1$, contradiction. $\square$

## Finite Fields

**Theorem A.** Suppose $p$ prime, $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$. Let $f(X) \in \mathbb{F}_p[X]$ be irreducible. Then $K = \mathbb{F}_p[X]/(f(X))$ is a field of order $p^{\deg f}$ and every finite field arises in this way.

**Theorem B.** Let $q = p^r$ be a prime power. Then there exists a field $\mathbb{F}_q$ of order $q$ and it is unique up to isomorphism.

**Theorem C.** The multiplicative group $\mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\}$ is cyclic, i.e. there exists $\beta \in \mathbb{F}_q$ such that $\mathbb{F}_q = \{0, 1, \beta, \dots, \beta^{q-2}\}$.

## BCH Codes

Let $n$ be an odd integer. Pick $r \ge 1$ such that $2^r \equiv 1 \pmod{n}$. (This exists since $(2,n) = 1$.) Let $K = \mathbb{F}_{2^r}$. Let $\mu_n(K) = \{x \in K : x^n = 1\} \le K^*$. Since $n \mid (2^r - 1) = |K^*|$, $\mu_n(K)$ is a cyclic group of order $n$. So $\mu_n(K) = \{1, \alpha, \dots, \alpha^{n-1}\}$ for some $\alpha \in K$, is called a *primitive nth root of unity*.

**Definition.** The cyclic code of length $n$ with defining set $A \subset \mu_n(K)$ is

$$C = \{f(X) \pmod{X^n - 1} : f(X) \in \mathbb{F}_2[X], f(a) = 0 \ \forall a \in A\}.$$

The generator polynomial is the non-zero polynomial $g(X)$ of least degree such that $g(a) = 0$ for all $a \in A$. Equivalently, $g(X)$ is the least common multiple of the minimal polynomials of the elements $a \in A$.

**Definition.** The cyclic code with defining set $A = \{\alpha, \alpha^2, \dots, \alpha^{\delta-1}\}$ is called a BCH (Bose, Ray-Chaudhuri, Hocquenghem) code with design distance $\delta$.

**Theorem 4.16.** A BCH code $C$ with design distance $\delta$ has $d(C) \geq \delta$.

**Lemma 4.17** (Vandermonde determinant)**.**

$$\begin{vmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ x_1^2 & x_2^2 & \dots & x_n^2 \\ \hdotsfor{4} \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{vmatrix} = \prod_{1 \leq j < i \leq n} (x_i - x_j)$$

*Proof.* This is an indentity in $\mathbb{Z}[X_1, \dots, X_n]$. The LHS vanishes when we specialise to $x_i = x_j$ for $i \neq j$. Therefore, $(x_i - x_j) \mid LHS$ for $i \neq j$.

Running over distinct permutations of $(i, j)$ we get coprime polynomials, so $RHS \mid LHS$. Both sides have degree $\binom{n}{2}$ and the coefficient of $x_2 x_3^2 \cdots x_n^{n-1}$ is 1 on the LHS and on the RHS. (On the RHS, we need to take a term with larger index from each bracket, so always take $x_i$, not $x_j$, whence the coefficient is 1.) Therefore, $LHS = RHS$. □

*Proof of Theorem 4.16.* Let

$$H = \begin{pmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \hdotsfor{5} \\ 1 & \alpha^{\delta-1} & \alpha^{2(\delta-1)} & \dots & \alpha^{(\delta-1)(n-1)} \end{pmatrix}$$

By Lemma 4.17, any $\delta - 1$ columns of $H$ are linearly independent. But any codeword of $C$ is a dependence relation between the columns of $H$. Hence every non-zero codeword has weights at least $\delta$. Therefore, $d(C) \geq \delta$. □

**Note 14.** $H$ is not a parity check matrix, its entries are not in $\mathbb{F}_2$.

## Decoding BCH Codes

Let $C$ be a cyclic code with defining set $\{\alpha, \alpha^2, \dots, \alpha^{\delta-1}\}$, where $\alpha \in K$ is a primitive $n$th root of unity. By Theorem 4.16, we ought to be able to correct $t = \left\lfloor \frac{\delta-1}{2} \right\rfloor$ errors. We send $c \in C$ and receive $r = c + e$, where $e$ is the error pattern. Note here

$$\begin{aligned} \mathbb{F}_2^n &\longleftrightarrow \mathbb{F}_2[X]/(X^n - 1) \\ r, c, e &\longleftrightarrow r(X), c(X), e(X) \end{aligned}$$

**Definition.** The *error locator polynomial* is

$$\sigma(X) = \prod_{i \in \mathcal{E}} (1 - \alpha^i X) \in K[X]$$

where $\mathcal{E} = \{0 \leq i \leq n-1 : e_i = 1\}$.

**Theorem 4.18.** Assume $\deg\sigma = |\mathcal{E}| \le t$. Then $\sigma(X)$ is the unique polynomial in $K[X]$ of least degree such that

(i) $\sigma(0) = 1$;
(ii) $\sigma(X)\sum_{j=1}^{2t} r(\alpha^j)X^j \equiv \omega(X) \pmod{X^{2t+1}}$ for some $\omega(X) \in K[X]$ with $\deg\omega \le t$.

*Proof.* Let $\omega(X) = -X\sigma'(X)$. Then

$$\omega(X) = \sum_{i\in\mathcal{E}} \alpha^i X \prod_{\substack{j\in\mathcal{E}\\ j\neq i}} (1-\alpha^j X).$$

We work in the power series ring $K[[X]]$.

$$\begin{aligned}\frac{\omega(X)}{\sigma(X)} &= \sum_{i\in\mathcal{E}} \frac{\alpha^i X}{1-\alpha^i X} \\ &= \sum_{i\in\mathcal{E}}\sum_{j=1}^{\infty} (\alpha^i X)^j \\ &= \sum_{j=1}^{\infty}\left(\sum_{i\in\mathcal{E}} (\alpha^j)^i\right) X^j \\ &= \sum_{j=1}^{\infty} e(\alpha^j)X^j\end{aligned}$$

Therefore,

$$\sigma(X)\sum_{j=1}^{\infty} e(\alpha^j)X^j = \omega(X).$$

By definition of $C$, $c(\alpha^j) = 0$ for all $1 \le j \le \delta - 1$. But $r = c + e$, so $r(\alpha^j) = e(\alpha^j)$ for all $1 \le j \le 2t$. Therefore,

$$\sigma(X)\sum_{j=1}^{2t} r(\alpha^j)X^j \equiv \omega(X) \pmod{X^{2t+1}}.$$

We have checked (i) and (ii) with $\omega(X) = -X\sigma'(X)$, so $\deg\omega = \deg\sigma = |\mathcal{E}| \le t$.

Suppose $\tilde{\sigma}(X), \tilde{\omega}(X) \in K[X]$ also satisfy (i), (ii) and $\deg\tilde{\sigma} \le \deg\sigma$. Note if $i \in \mathcal{E}$,

$$\omega(\alpha^{-i}) = \prod_{\substack{j\in\mathcal{E}\\ j\neq i}} (1-\alpha^{j-i}) \neq 0$$

so $\sigma(X)$ and $\omega(X)$ are coprime. By (ii),

$$\sigma(X)\tilde{\omega}(X) \equiv \tilde{\sigma}(X)\omega(X) \pmod{X^{2t+1}},$$

so $\sigma(X)\tilde{\omega}(X) = \tilde{\sigma}(X)\omega(X)$ since $\sigma$, $\tilde{\sigma}$, $\omega$, $\tilde{\omega}$ all have degree at most $t$.

But $\sigma(X), \omega(X)$ are coprime, so $\sigma(X) \mid \tilde{\sigma}(X)$. We assumed $\deg\tilde{\sigma} \le \deg\sigma$, so $\tilde{\sigma} = a\sigma$ for some $a \in K$. Then by (i), $\tilde{\sigma} = \sigma$. $\square$

### Decoding algorithm

Suppose we receive the word $r(X)$.

(i) Compute $\sum_{j=0}^{2t} r(\alpha^j)x^j$.
(ii) Set $\sigma(X) = 1 + \sigma_1 X + \cdots + \sigma_t X^t$ and compare coefficients of $X^i$ for $t+1 \leq i < 2t$ to obtain linear equations for $\sigma_1, \ldots, \sigma_t$.
(iii) Solve these over $K$, e.g. using Gaussian elimination, keeping solutions of least degree.
(iv) Compute $\mathcal{E} = \{0 \leq i \leq n-1 : \sigma(\alpha^{-i} = 0\}$ and check $|\mathcal{E}| = \deg \sigma$.
(v) Set $e(X) = \sum_{i \in \mathcal{E}} X^i$, $c(X) = r(X) + e(X)$ and check $c(X)$ is a codeword.

**Example.** (i) Let $n = 7$. $X^7 - 1 = (X-1)(X^3 + X + 1)(X^3 + X^2 + 1)$ in $\mathbb{F}_2[X]$. For example, take $g(X) = X^3 + X + 1$ and $h(X) = (X+1)(X^3 + X^2 + 1) = X^4 + X^2 + X + 1$. The parity check matrix is

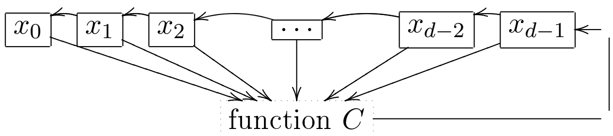
$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

This is the Hamming $(7,4)$-code.

(ii) Let $K$ be a splitting field of $X^7 - 1 \in \mathbb{F}_2[X]$, e.g. $K = \mathbb{F}_8$. Let $\beta \in K$ be a root of $g(X)$. Therefore, $\beta$ is a primitive 7th root of unity. Note $\beta^3 = \beta + 1$, so $\beta^6 = (\beta+1)^2 = \beta^2 + 1$, so $g(\beta^2) = 0$. Therefore, the BCH code $C$ defined by $\{\beta, \beta^2\}$ has generator polynomial $g(X)$, it is Hamming's $(7,4)$-code again. So by Theorem 4.16, $d(C) \geq 3$.

## Shift Registers

**Definition.** A *(general) feedback shiftback register* is a function $f \colon \mathbb{F}_2^d \to \mathbb{F}_2^d$ of the form $f(x_0, x_1, \ldots, x_{d-1}) = (x_1, \ldots, x_{d-1}, C(x_0, \ldots, x_{d-1}))$ for some function $C \colon \mathbb{F}_2^d \to \mathbb{F}_2$. We say the register has *length* $d$.




The register is *linear* (LFSR) if $C$ is a linear map, say $(x_0, \ldots, x_{d-1}) \mapsto \sum_{i=0}^{d-1} a_i x_i$.

The initial fill $(y_0, y_1, \ldots, y_{d-1})$ produces an output sequence $(y_n)_{n \geq 0}$ given by

$$\begin{aligned} y_{n+d} &= C(y_n, y_{n+1}, \ldots, y_{n+d-1}) \\ &= \sum_{i=0}^{d-1} a_i y_{n+i} \end{aligned}$$

i.e. we have a sequence determined by a linear recurrence relation with *auxiliary polynomial* $P(X) = X^d + a_{d-1}X^{d-1} + \cdots a_1 X + a_0$.

**Definition.** The *feedback polynomial* is $\tilde{P}(X) = a_0 X^d + a_1 X^{d-1} + \cdots + a_{d-1}X + 1$.

**Lemma 4.19.** The sequence $(y_n)_{n\geq 0}$ in $\mathbb{F}_2$ is the output from a LSFR with auxiliary polynomial $P(X)$ if and only if

$$\sum_{i=0}^{\infty} y_i X^i = \frac{A(X)}{\tilde{P}(X)}$$

for some $A(X) \in \mathbb{F}_2[X]$, with $\deg A < \deg P$ and $\tilde{P}(X) = X^{\deg P} P(X^{-1}) \in \mathbb{F}_2[X]$.

*Proof.* Let $P(X) = a_d X^d + \cdots + a_1 X + a_0$. Therefore, $\tilde{P}(X) = a_0 X^d + \cdots + a_{d-1} X + a_d$. The condition is that $\left(\sum_{i=0}^{\infty} y_i X^i\right)\tilde{P}(X)$ is a polynomial of degree less than $d$. This holds if and only if

$$\begin{aligned} &\sum_{i=0}^{d-1} a_i y_{n-d+1} = 0 \quad \forall n \geq d \\ \iff &\sum_{i=0}^{d-1} a_i y_{n+i} = 0 \quad \forall n \geq 0 \end{aligned}$$

if and only if $(y_n)_{n\geq 0}$ is the output from a LSFR. □

The following problems are closely related.

(i) Decoding BCH codes (see Theorem 4.18);
(ii) recovering a LFSR from its output stream (see Lemma 4.19);
(iii) writing a power series as a ratio of polynomials.

## Berlekamp Massey Method

Let $(x_n)_{n\geq 0}$ be the output from a LFSR. Our aim is to find $d$ and $a_0, \ldots, a_{d-1} \in \mathbb{F}_2$ such that $x_{n+d} = \sum_{i=0}^{d-1} a_i x_{n+i}$ for all $n \geq 0$. We have

$$\underbrace{\begin{pmatrix} x_0 & x_1 & \cdots & x_d \\ x_1 & x_2 & \cdots & x_{d+1} \\ \cdots & \cdots & \cdots & \cdots \\ x_d & x_{d+1} & \cdots & x_{2d} \end{pmatrix}}_{=:A_d} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{d-1} \\ 1 \end{pmatrix} = 0 \qquad (*)$$

If we know that the register has length at least $r$, start with $i = r$. Compute $\det A_i$.

- If $\det A_i \neq 0$, then $d > i$, replace $i$ by $i + 1$ and repeat.
- If $\det A_i = 0$, solve $(*)$ for $a_0, \ldots, a_{d-1}$ by Gaussian elemination and test the solution over as many terms of the sequence as we like. If it fails, then $d > i$, replace $i$ by $i + 1$ and repeat.

# Chapter 5

# Cryptography

The aim is to modify the message such that it is unintelligible to an eavesdropper.

There is some secret information shared by the sender and receiver, called the *key* in $\mathfrak{K}$. The unencrypted message is called the *plaintext* and from $\mathfrak{M}$. The encrypted message is called the *ciphertext* and it is from $\mathfrak{C}$. A cryptosystem consists of sets $(\mathfrak{K}, \mathfrak{M}, \mathfrak{C})$ with functions

$$\begin{aligned} e &\colon \mathfrak{M} \times \mathfrak{K} \to \mathfrak{C} \\ d &\colon \mathfrak{C} \times \mathfrak{K} \to \mathfrak{M} \end{aligned}$$

such that $d(e(m,k),k) = m$ for all $m \in \mathfrak{M}, k \in \mathfrak{K}$.

**Example.** Some examples in the case $\mathfrak{M} = \mathfrak{C} = \Sigma = \{A, B, \ldots, Z\}$.

(i) Simple substitution, $\mathfrak{K}$ is the set of permutations of $\Sigma$. Each letter of the plaintext is replaced by the image under the permutation.
(ii) Vigenère cipyer. $\mathfrak{K} = \Sigma^d$ for some $d \in \mathbb{N}$. Identify $\Sigma$ and $\mathbb{Z}/26\mathbb{Z}$. Write out the key repeatedly below the message and add modulo 26.

What does it mean to break a cryptosystem? The enemy might know

- the functions $d$ and $e$,
- the probability distributions on $\mathfrak{M}, \mathfrak{K}$,

but not the key. They seek to recover the plaintext from the ciphertext.

There are three possible attacks.

1. Ciphertext only. The enemy knows some piece of the ciphertext.
2. Known plaintext. The enemy possesses a considerable length of plaintext and matching ciphertext, and seeks to discover the key.
3. Chosen plaintext. The enemy may aquire the ciphertext for any message he chooses.

Examples (i) and (ii) fail at the level 2, at least for sufficiently random messages. They even fail at level 1, if e.g. the source is English text. For modern applications, level 3 is desirable.

We model the key and the messages as independent random variables $K$ and $M$ taking values in $\mathfrak{K}$ and $\mathfrak{M}$. Put $C = e(K, M)$.

**Definition.** A cryptosystem has *perfect secrecy* if $M$ and $C$ are independent. Equivalently, $I(M; C) = 0$.

**Lemma 5.1.** Perfect secrecy implies $|\mathfrak{K}| \geq |\mathfrak{M}|$.

*Proof.* Pick $m_0 \in \mathfrak{M}$ and $k_0 \in \mathfrak{K}$ with $\mathbb{P}(K = k_0) > 0$. Let $c_0 = e(m_0, k_0)$. For any $m \in \mathfrak{M}$,

$$\mathbb{P}(C = c_0 \mid M = m) = \mathbb{P}(C = c_0) = \mathbb{P}(C = c_0 \mid M = m_0) = \mathbb{P}(K = k_0) > 0.$$

So for each $m \in \mathfrak{M}$ there exists $k \in \mathfrak{K}$ such that $e(m, k) = c_0$. Therefore, $|\mathfrak{K}| \geq |\mathfrak{M}|$. □

We conclude that perfect secrecy is an unrealistic goal.

**Definition.** (i) The *message equivocation* is $H(M \mid C)$.
(ii) The *key equivocation* is $H(K \mid C)$.

**Lemma 5.2.** $H(M \mid C) \leq H(K \mid C)$.

*Proof.* Since $M = d(C, K)$, $H(M \mid C, K) = 0$. So $H(C, K) = H(M, C, K)$. Therefore,

$$\begin{aligned}
H(K \mid C) &= H(M, C, K) - H(C) \\
&= H(K|M, C) + H(M, C) - H(C) \\
&= \underbrace{H(K \mid M, C)}_{\geq 0} + H(M \mid C) \\
\therefore \quad H(K \mid C) &\geq H(M, C)
\end{aligned}$$

□

Take $\mathfrak{M} = \mathfrak{C} = \Sigma$, say. We send $n$ messages $M^{(n)} = (M_1, \dots, M_n)$ encrypted as $C^{(n)} = (C_1, \dots, C_n)$ using the same key.

**Definition.** The *unicity distance* is the least $n$ for which $H(K \mid C^{(n)}) = 0$, i.e. the smallest number of encrypted messages required to uniquely determine the key.

$$\begin{aligned}
H(K \mid C^{(n)}) &= H(K, C^{(n)}) - H(C^{(n)}) \\
&= H(K, M^{(n)}) - H(C^{(n)}) \\
&= H(K) + H(M^{(n)}) - H(C^{(n)}).
\end{aligned}$$

We assume that

(i) all keys are equally likely, so $H(K) = \log|\mathfrak{K}|$;
(ii) $H(M^{(n)}) \approx nH$ for some constant $H$, for sufficiently large $n$ (this is true for many sources, including Bernoulli sources);
(iii) all sequences of ciphertext are equally likely, so $H(C^{(n)}) = n \log|\Sigma|$ (good cryptosystems should satisfy this).

So

$$\begin{aligned}
H(K \mid C^{(n)}) &= \log|\mathfrak{K}| + nH - n\log|\Sigma| \\
&\geq 0
\end{aligned}$$

if and only if

$$n \leq U := \frac{\log|\mathfrak{K}|}{\log|\Sigma| - H}$$

which is the unicity distance.

Recall that $0 \leq H \leq \log|\Sigma|$. To make the unicity distance large we can make $\mathfrak{K}$ large or use a message source with little redundancy.

**Example.** Suppose we can decrypt a substitution cipher after 40 letters. $|\Sigma| = 26, |K| = 26!, U \leq 40$. Then for the entropy of English text $H_E$ we have

$$H_E \leq \log 26 \frac{\log 26!}{40} \approx 2.5$$

Many cryptosystems are thought secure (and indeed used) beyond the unicity distance.

## Stream Ciphers

We work with streams, i.e. sequences in $\mathbb{F}_2$. For plaintext $p_0, p_1, \dots$ and key $k_0, k_1, \dots$ we set the ciphertext to be $z_0, z_1, \dots$ where $z_n = p_n + k_n$.

### One time pad

The key stream is a random sequence, known only to the sender and recipient. Let $K_0, K_1, \dots$ be i.i.d. random variables with $\mathbb{P}(K_j = 0) = \mathbb{P}(K_j = 1) = \frac{1}{2}$. The ciphertext is $Z_n = p_n + K_n$, where the plaintext is fixed. Then $Z_0, Z_1, \dots$ are i.i.d. random variables with $\mathbb{P}(Z_j = 0) = \mathbb{P}(Z_j = 1) = \frac{1}{2}$. Therefore, without knowledge of the key stream deciphering is impossible. (Hence this has infinite unicity distance.)

There are the following two problems with the use of one time pads.

(i) How do we construct a random key sequence?
(ii) How do we share the key sequence?

(i) is surprisingly tricky, but not a problem in practice. (ii) is the same problem we started with. In most applications, the one time pad is not practical. Instead, we generate $k_0, k_1, \dots$ using a feedback shift register, say of length $d$. We only need to share the initial fill $k_0, k_1, \dots, k_{d-1}$.

**Lemma 5.3.** Let $x_0, x_1, \dots$ be a stream produced by a shift register of length $d$. Then there exist $M, N \leq 2^d$ such that $x_{N+r} = x_r$ for all $r \geq M$.

*Proof.* Let the register be $f\colon \mathbb{F}_2^d \to \mathbb{F}_2^d$. Let $v_i = (x_i, x_{i+1}, \dots, x_{i+d-1})$. Then $v_{i+1} = f(v_i)$. Since $|\mathbb{F}_2^d| = 2^d$, the vectors $v_0, v_1, \dots, v_{2^d}$ cannot all be distinct, so there exist $0 \leq a < b \leq 2^d$ such that $v_a = v_b$. Let $M = a, N = b - a$. So $v_M = v_{M+N}$ and $v_r = v_{r+N}$ for all $r \geq M$ (by induction, apply $f$), so $x_r = x_{r+N}$ for all $r \geq M$. $\square$

**Remark.** (i) The maximum period of a feedback shift register of length $d$ is $2^d$.
(ii) The maximum period of a LFSR of length $d$ is $2^d - 1$. The bound of Lemma 5.3 is improved by 1, since we can assume $v_i \neq 0$ for all $i$, otherwise the period is 1. But we can obtain period $2^d - 1$ by taking $x_n = T(\alpha^n)$ where $\alpha$ is a generator for $\mathbb{F}_{2^d}^*$ and $T\colon \mathbb{F}_{2^d} \to \mathbb{F}_2$ is any non-zero $\mathbb{F}_2$-linear map. We must check that $(x_n)$ is the output from a LFSR and the sequence does not repeat itself with period less than $2^d - 1$ (see Example Sheet 4).
(iii) Stream ciphers using a LSFR fail at level 2 (known plaintext attack), due to the Berlekamp Massey method.

Why should this cryptosystem be used?

(i) It is cheap, fast and easy to use.
(ii) Messages are encrypted and decrypted "on the fly".
(iii) It is error-tolerant.

**Solving linear recurrence relations**

Recall that over $\mathbb{C}$ the general solution is a linear combination of solutions $\alpha^n$, $n\alpha^n$, $n^2\alpha^n$, ..., $n^{t-1}\alpha^n$ for $\alpha$ a root of the auxiliary polynomial $P(X)$ with multiplicity $t$. Beware that $n^2 \equiv n \pmod 2$. Over $\mathbb{F}_2$, we need two modifications.

(i) We work in a splitting field $K$ for $P(X) \in \mathbb{F}_2[X]$;
(ii) replace $n^i\alpha^n$ by $\binom{n}{i}\alpha^n$.

We can also generate new key streams from old ones as follows.

**Lemma 5.4.** Let $x_n$ and $y_n$ be the output from a LFSR of length $M$ and $N$, respectively.

(i) The sequence $(x_n + y_n)$ is the output from a LFSR of length $M + N$.
(ii) The sequence $(x_n y_n)$ is the output from a LFSR of length $MN$.

*Proof.* We will assume that the auxiliary polynomials $P(X), Q(X)$ each have distinct roots, say $\alpha_1, \dots, \alpha_M$ and $\beta_1, \dots, \beta_N$ in some extension field $K$ of $\mathbb{F}_2$. Then $x_n = \sum_{i=1}^{M} \lambda_i \alpha_i^n$, $y_n = \sum_{j=1}^{N} \mu_j \beta_j^n$ for some $\lambda_i, \mu_j \in K$.

(i) $x_n + y_n = \sum_{i=1}^{M} \lambda_i \alpha_i^n + \sum_{j=1}^{N} \mu_j \beta_j^n$. This is produced by a LFSR with auxiliary polynomial $P(X)Q(X)$.
(ii) $x_n y_n = \sum_{i=1}^{M} \sum_{j=1}^{N} \lambda_i \mu_j (\alpha_i \beta_j)^n$ is the output of a LFSR with auxiliary polynomial $\prod_{i=1}^{N} \prod_{j=1}^{M} (X - \alpha_i \beta_j)$, which is in $\mathbb{F}_2[X]$ by the Symmetric Function Theorem. □

We have the following conclusions.

(i) Adding the output of two LFSR is no more economical then producing the same string with a single LFSR.
(ii) Multiplying streams looks promising, until we realise that $x_n y_n = 0$ 75% of the time.

**Remark.** Non-linear registers look appealing, but are difficult to analyse. In particular, the eavesdropper may understand them better than we do.

**Example.** Take $x_n, y_n, z_n$ output from LFSRs. Put

$$k_n = \begin{cases} x_n & \text{if } z_n = 0 \\ y_n & \text{if } z_n = 1 \end{cases}$$

To apply Lemma 5.4, write $k_n = x_n + z_n(x_n + y_n)$ to deduce $(k_n)$ is again the output from a LFSR.

Stream ciphers are examples of *symmetric cryptosystems*, i.e. decryption is the same, or easily deduced from the encryption algorithm.

# Public Key Cryptography

This is an example of an asymmetric cryptosystem. We split the key into two parts.

- Private key for decryption.
- Public key for encryption.

Knowing the encryption and decryption algorithms and the public key, it should still be hard to find the private key or to decrypt messages. This aim implies security at level 3 (chosen plaintext). There is also no key exchange problem.

The idea is to base the system on mathematical problems that are believed to be hard. We consider two such problems.

(i) Factoring. Let $N = pq$ for $p, q$ large primes. Given $N$, find $p$ and $q$.
(ii) Discrete logarithms. Let $p$ be a large prime and $g$ be a primitive root modulo $p$, i.e. a generator for $\mathbb{F}_p^*$. Given $x$, find $a$ such that $x \equiv g^a \pmod p$.

**Definition.** An algorithm runs in polynomial time if

$$\#(\text{operations}) \leq c(\text{input size})^d$$

for some constants $c$ and $d$.

**Note 15.** An algorithm for factoring $N$ has input size $\log N$, i.e. the number of digits of $N$.

The following are polynomial time algorithms.

- Arithmetic of integers ($+$, $-$, $\times$, division algorithm);
- computation of GCD using Euclid's algorithm;
- modular exponentiation, i.e. computation of $x^\varphi \pmod N$ using the repeated squaring algorithm;
- primality testing (Agrawal, Kayal, Saxena 2002).

Polynomial time algorithms are not known for (i) and (ii).

### Elementary methods

(i) Trial division properly organised takes time $\mathcal{O}(\sqrt{N})$.
(ii) Baby-step Giant-step algorithm. Set $m = \lceil \sqrt{p} \rceil$, write $a = qm + r$, $0 \leq q, r < m$. Then

$$\begin{aligned} x \equiv g^a \equiv g^{qm+r} &\pmod p \\ \therefore \quad g^{qm} \equiv g^{-r} x &\pmod p \end{aligned}$$

List $g^{qm} \pmod p$ for $q = 0, 1, \ldots, m-1$ and $g^{-r}x \pmod p$ for $r = 0, 1, \ldots, m-1$. Sort these two lists and look for a match. Therefore, we can find discrete logarithms in time and storage $\mathcal{O}(\sqrt{p}\log p)$.

### Factor base

Let $\mathcal{B} = \{q \text{ prime} : q \le C\} \cup \{-1\}$ for some constant $C$.

(i) Find relations of the form $x^2 \equiv \prod_{q\in\mathcal{B}} q^{\alpha(q,x)} \pmod{N}$. Linear algebra over $\mathbb{F}_2$ allows us to multiply such relations together to obtain $x^2 \equiv y^2 \pmod{N}$, hence $(x-y)(x+y) \equiv 0 \pmod{N}$. Taking $\gcd(x-y, N)$ may give a non-trivial factor of $N$, repeat otherwise.

(ii) Find relations of the form $g^r \equiv \prod_{q\in\mathcal{B}} q^{\alpha(q,r)} \pmod{p}$. With enough relations, solving linear equations modulo $p-1$ will solve the discrete logarithm problem for each $q \in \mathcal{B}$. Then find $s$ such that $xg^s \equiv \prod_{q\in\mathcal{B}} q^{\beta(q,s)} \pmod{p}$. Therefore, we can solve the discrete logarithm problem for $x$.

The best known method for solving (i) and (ii) uses a factor base method called the number field sieve. It has running time $\mathcal{O}(e^{c(\log N)^{1/3}(\log\log N)^{2/3}})$ where $c$ is a known constant. Note this is closer to polynomial time (in $\log N$) than to exponential time (in $\log N$) tanks to the exponents $\frac{1}{3}$ and $\frac{2}{3}$.

RSA factoring challenges.

| | # decimal digits | Factored | Price money |
|---|---|---|---|
| RSA-576 | 174 | 3rd Dec 2003 | \$10,000 |
| RSA-640 | 193 | 2nd Nov 2005 | \$20,000 |
| RSA-704 | 212 | Not factored | \$30,000 |

Recall that

$$\begin{aligned}\phi(n) &= |\{1 \le a \le n : (a,n) = 1\}| \\ &= |(\mathbb{Z}/n\mathbb{Z})^*|,\end{aligned}$$

the number of units in $\mathbb{Z}/n\mathbb{Z}$. The Euler-Fermat theorem states

$$(a,n) = 1 \implies a^{\phi(n)} \equiv 1 \pmod{n}.$$

A special case of this is Fermat's little theorem, stating that for prime $p$

$$(a,p) = 1 \implies a^{p-1} \equiv 1 \pmod{p}.$$

**Lemma 5.5.** Let $p = 4k-1$ be prime, $d \in \mathbb{Z}$. If $x^2 \equiv d \pmod{p}$ is soluble then a solution is $x \equiv d^k \pmod{p}$.

*Proof.* Let $x_0$ be a solution. Without loss of generality, we may assume $x_0 \not\equiv 0 \pmod{p}$. Then

$$\begin{aligned} & d^{2k-1} \equiv x_0^{2(2k-1)} \equiv x_0^{p-1} \equiv 1 \pmod{p} \\ \therefore \quad & (d^k)^2 \equiv d \pmod{p}. \end{aligned}$$ □

### Rabin Williams cryptosystem

The private key consists of two large distinct primes $p, q \equiv 3 \pmod 4$. The public key is $N = pq$. We have $\mathfrak{M} = \mathfrak{C} = \{0, 1, 2, \ldots, N-1\}$. We encrypt a message $m \in \mathfrak{M}$ as $c = m^2 \pmod N$. The ciphertext is $c$. (We should avoid $m < \sqrt{N}$.)

Suppose we receive $c$. Use Lemma 5.5 to solve for $x_1, x_2$ such that $x_1^2 \equiv c \pmod p$, $x_2^2 \equiv c \pmod q$. Then use the Chinese Remainder Theorem (CRT) to find $x$ with $x \equiv x_1 \pmod p$, $x \equiv x_2 \pmod q$, hence $x^2 \equiv c \pmod N$. Indeed, running Euclid's algorithm on $p$ and $q$ gives integers $r, s$ with $rp + sq = 1$. We take $x = (sq)x_1 + (rp)x_2$.

**Lemma 5.6.**
(i) Let $p$ be an odd prime and $\gcd(d, p) = 1$. Then $x^2 \equiv d \pmod p$ has no or two solutions.
(ii) Let $N = pq$, $p, q$ distinct odd primes and $\gcd(d, N) = 1$. Then $x^2 \equiv d \pmod N$ has no or four solutions.

*Proof.* (i)

$$\begin{aligned} & x^2 \equiv y^2 \pmod p \\ \iff & p \mid (x+y)(x-y) \\ \iff & p \mid (x+y) \text{ or } p \mid (x-y) \\ \iff & x \equiv \pm y \pmod p. \end{aligned}$$

(ii) If $x_0$ is some solution, then by CRT there exist solutions $x$ with $x \equiv \pm x_0 \pmod p$, $x \equiv \pm x_0 \pmod q$ for any of the four choices of $\pm$. By (i), these are the only solutions. □

To decrypt Rabin Williams, we find all four solutions to $x^2 \equiv c \pmod N$. Messages should include enough redundancy that only one of these possibilities makes sense.

**Theorem 5.7.** Breaking the Rabin Williams cryptosystem is essentially as difficult as factoring $N$.

*Proof.* We have seen that factoring $N$ allows us to decrypt messages. Conversely, suppose we have an algorithm for computing square roots modulo $N$. Pick $x \pmod N$ at random. Use the algorithm to find $y$ such that $y^2 \equiv x^2 \pmod N$. With probability $\frac{1}{2}$, $x \not\equiv y \pmod N$. Then $\gcd(N, x-y)$ is a non-trivial factor of $N$. If this fails, start again with another $x$. After $r$ trials, the probability of failure is less that $\frac{1}{2^r}$, which becomes arbitrarily small. □

Let $N = pq$, $p, q$ distinct odd primes. We show that if we know a multiple $m$ of $\phi(N) = (p-1)(q-1)$ then factoring $N$ is easy.

**Notation.** Let $o_p$ be the order of $x$ in $(\mathbb{Z}/p\mathbb{Z})^*$. Write $m = 2^a b$, $a \geq 1$, $b$ odd. Let

$$X = \{x \in (\mathbb{Z}/N\mathbb{Z})^* : o_p(x^b) = o_q(x^b)\}.$$

**Theorem 5.8.**
(i) If $x \in X$ then there exists $0 \leq t < a$ such that $\gcd(x^{2^t b} - 1, N)$ is a non-trivial factor of $N$.
(ii) $|X| \geq \frac{1}{2}|(\mathbb{Z}/N\mathbb{Z})^*| = \frac{\phi(N)}{2}$.

*Proof.* (i) By Euler-Fermat,

$$\begin{aligned} x^{\phi(N)} &\equiv 1 \pmod{N} \\ \implies \quad x^m &\equiv 1 \pmod{N}. \end{aligned}$$

But $m = 2^a b$, so putting $y = x^b \pmod{N}$ we get $y^{2^a} \equiv 1 \pmod{N}$. Therefore, $o_p(y)$ and $o_q(y)$ are powers of 2. We are given $o_p(y) \neq o_q(y)$, and without loss of generality we may assume $o_p(y) < o_q(y)$. Say $o_p(y) = 2^t$, so $0 \le t < a$. Then

$$\begin{aligned} y^{2^t} &\equiv 1 \pmod{p} \\ y^{2^t} &\not\equiv 1 \pmod{q} \end{aligned}$$

So $\gcd(y^{2^t} - 1, N) = p$.

(ii) See Page 52. $\square$

**RSA (Rivest, Shamir, Adleman)**

Let $N = pq$, $p, q$ large distinct primes. Recall that $\phi(N) = (p-1)(q-1)$. Pick $e$ with $\gcd(e, \phi(N)) = 1$. We solve for $d$ such that $de \equiv 1 \pmod{\phi(N)}$.

The public key is $(N, e)$, the private key is $(N, d)$.

We encrypt $m \in \mathfrak{M}$ as $c = m^2 \pmod{N}$ and decrypt $c$ as $x = c^d \pmod{N}$. By Euler-Fermat, $x = m^{de} \equiv m \pmod{N}$ since $de \equiv 1 \pmod{\phi(N)}$. (We ignore the possibility that $\gcd(m, N) \neq 1$, since this occurs with very small probability.)

**Corollary 5.9.** Finding the RSA private key $(N, d)$ from the public key $(N, e)$ is essentially as difficult as factoring $N$.

*Proof.* We have seen that factoring $N$ allows us to find $d$. Conversely, if we know $d$ and $e$, $de \equiv 1 \pmod{\phi(N)}$, then $\phi(N) \mid (de - 1)$ from taking $m = de - 1$ in Theorem 3.13. $\square$

*Proof of Theorem 5.8 (ii).* By the CRT we have the following correspondence.

$$\begin{aligned} (\mathbb{Z}/N\mathbb{Z})^* &\longrightarrow (\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^* \\ x &\longmapsto (x \mod p, x \mod q) \end{aligned}$$

It suffices to show that if we partition $(\mathbb{Z}/p\mathbb{Z})^*$ according to the value of $o_p(x^b)$ then each subset has size at most $\frac{1}{2}|(\mathbb{Z}/p\mathbb{Z})^*| = \frac{p-1}{2}$. We show that some subset has size $\frac{1}{2}|(\mathbb{Z}/p\mathbb{Z})^*|$. Recall that $(\mathbb{Z}/p\mathbb{Z})^* = \{1, g, g^2, \dots, g^{p-1}\}$. By Fermat's little theorem,

$$\begin{aligned} g^{p-1} &\equiv 1 \pmod{p} \\ \therefore \quad g^{2^a b} &\equiv 1 \pmod{p} \end{aligned}$$

and hence $o_p(g^b)$ is a power of 2. So

$$o_p(g^{b\delta}) \begin{cases} = o_p(g^b) & \text{if } \delta \text{ odd} \\ < o_p(g^b) & \text{otherwise} \end{cases}$$

Therefore, $\{g^\delta \mod p : \delta \text{ odd}\}$ is the required set. $\square$

**Remark.** It is not known whether decrypting RSA messages without knowledge of the private key is essentially as hard as factoring.

### Diffie–Hellman key exchange

Let $p$ be a large prime, $g$ a primitve root modulo $p$. This data is fixed and known to everyone.

Alice and Bob wish to agree a secret key. $A$ chooses $\alpha \in \mathbb{Z}$ and sends $g^\alpha \pmod p$ to $B$. $B$ chooses $\beta \in \mathbb{Z}$ and sends $g^\beta \pmod p$ to $A$. They both compute $k = (g^\beta)^\alpha = (g^\alpha)^\beta \pmod p$ and use this as their secret key.

The eavesdropper seeks to compute $g^{\alpha\beta}$ from $g, g^\alpha, g^\beta, p$. This is conjectured, although not proven, to be as hard as the discrete logarithm problem.

## Authentication and Signatures

Alice sends a message to Bob. Possible aims include the following.

- Secrecy. $A$ and $B$ can be sure that no third party can read the message.
- Integrity. $A$ and $B$ can be sure that no third party can alter the message.
- Authenticity. $B$ can be sure that $A$ sent the message.
- Non-repudiation. $B$ can prove to a third party that $A$ sent the message.

### Authentication using RSA

$A$ uses the private key $(N, d)$ to encrypt messages. Anyone can decrypt messages using the public key $(N, e)$. (Note that $(x^d)^e = (x^e)^d \equiv x$.) But they cannot forge messages sent by $A$.

### Signatures

Signature schemes can be used to preserve integrity and non-repudiation. They also prevent tampering of the following kind.

**Example** (Homomorphism attack)**.** A bank sends messages of the form $(M_1, M_2)$ where $M_1$ is the name of the client and $M_2$ is the amount transferred to his account. Messages are encoded using RSA

$$(Z_1, Z_2) = (M_1^e \mod N, M_2^e \mod N).$$

I transfer £100 to my account, observe the encrypted message $(Z_1, Z_2)$ and then send $(Z_1, Z_2^3)$. I become a millionaire without the need to break RSA.

**Example** (Copying)**.** I could just keep sending $(Z_1, Z_2)$. This is defeated by time stamping.

A message $m$ is signed as $(m, s)$ where $s$ is a function of $m$ and the private key. The signature (or trapdoor) function should be designed so no-one without knowledge of the private key can sign messages, yet anyone can check the signature is valid.

**Remark.** We are interested in the signature of the message, not of the sender.

### Signatures using RSA

$A$ has private key $(N, d)$, public key $(N, e)$. She signs $m$ as $(m, m^d \mod N)$. The signature $s$ is verified by checking $s^e \equiv m \pmod N$.

There are the following problems.

(i) The homomorphism attack still works.

(ii) Existential forgery. Anyone can produce valid signed messages of the form $(s^e \mod N, s)$ after choosing $s$ first. We might hope that messages generated in this way are not meaningful.

However, there are the following solutions.

(i) We can use a better signature scheme, as explained later.

(ii) Rather than signing the message $m$, we sign $h(m)$ where $h$ is a hash function. $h\colon \mathfrak{M} \to \{0, 1, \ldots, N-1\}$ is a publically known function for which it is very difficult to find pairs $x, x' \in \mathfrak{M}$ with $x \neq x'$ and $h(x) = h(x')$.

### The el Gamal signature scheme

Let $p$ be a large prime, $g$ a primitive element modulo $p$. Alice randomly chooses an integer $u$, $1 \leq u \leq p-1$. The public key is $p, g, y = g^u \pmod p$. The private key is $u$.

To send a message $m$, $1 \leq m \leq p-1$, Alice randomly chooses $k$, coprime to $p-1$, and computes $r, s$ with $1 \leq r, s \leq p-1$ satisfying

$$r \equiv g^k \pmod p \tag{1}$$
$$m \equiv ur + ks \pmod{p-1} \tag{2}$$

Alice signs the message $m$ with signature $(r, s)$. Now

$$\begin{aligned} g^m &\equiv g^{ur+ks} \pmod p \quad \text{by } (2) \\ &\equiv (g^u)^r (g^k)^s \pmod p \\ &\equiv y^r r^s \pmod p \end{aligned}$$

Bob accepts the signature if $g^m \equiv y^r r^s \pmod p$.

How can we forge such a signature? All obvious attacks involve solving the discrete logarithm problem.

**Lemma 5.10.** Given $a, b, m$, the congruence

$$ax \equiv b \pmod m \tag{$*$}$$

has either zero or $\gcd(a, m)$ solutions.

*Proof.* Let $d = \gcd(a, m)$. If $d \nmid b$ then there are no solutions. Otherwise rewrite the congruence $(*)$ as

$$\frac{a}{d} x \equiv \frac{b}{d} \pmod{\frac{m}{d}} \tag{$**$}$$

Now $\gcd(\frac{a}{d}, \frac{m}{d}) = 1$, so $(**)$ has a unique solution modulo $\frac{m}{d}$, so $(*)$ has $d$ solutions modulo $m$. $\square$

It is important that Alice chooses a new value of $k$ to sign each message. Otherwise suppose messages $m_1, m_2$ have signatures $(r, s_1)$ and $(r, s_2)$.

$$\begin{aligned} m_1 &\equiv ur + ks_1 \pmod{p-1} \qquad (\dagger) \\ m_2 &\equiv ur + ks_2 \pmod{p-1} \\ \therefore \quad m_1 - m_2 &\equiv k(s_1 - s_2) \pmod{p-1} \end{aligned}$$

By Lemma 5.10, this congruence has $d = \gcd(s_1 - s_2, p-1)$ solutions for $k$. If $d$ is small, we run through all possibilities for $k$ and see which of them satisfy $r \equiv g^k \pmod{p}$. Now similarly, we use ($\dagger$) to solve for $u$. This is Alice's private key, so we can now sign messages.

**Remark.** Several existential forgeries are known, i.e. we can find solutions $m, r, s$ to $g^m \equiv y^r r^s \pmod{p}$, but with now control over $m$. In practice, this is stopped by signing a hash value of the message instead of the message itself.

## Bit Commitment

Alice would like to send a message to Bob in such a way that

(i) Bob cannot read the message until Alice sends further information;
(ii) Alice cannot change the message.

This has the following applications.

- Coin tossing;
- sell stock market tips;
- multiparty computation, e.g. voting, surveys, etc.

We now present two solutions.

(i) Using any public key cryptosystem. Bob cannot read the message until Alice sends her private key.
(ii) Using coding theory as follows.

$$\begin{tikzcd} \boxed{\text{Alice}} \arrow[rr, bend left, "\text{noisy channel}"] \arrow[rr, bend right, "\text{clear channel}"'] & & \boxed{\text{Bob}} \end{tikzcd}$$

The noisy channel is modelled as a BSC with error probability $p$. Bob chooses a linear code $C$ with appropriate parameters. Alice chooses a linear map $\phi\colon C \to \mathbb{F}_2$. To send $m \in \{0, 1\}$, Alice chooses $c \in C$ such that $\phi(c) = m$ and sends $c$ to Bob via the noisy channel. Bob receives $r = c + e$, $d(r, c) = \omega(e) \approx np$. (The variance of the BSC should be chosen small.) Later Alice sends $c$ via the clear channel and Bob checks $d(r, c) \approx np$.

Why can Bob not read the message? We arrange that $C$ has minimum distance much smaller than $np$.

Why can Alice not change her choice? Alice knows the codeword $c$ sent, but not $r$. If later she sends $c'$ it will only be accepted if $d(c', r) \approx np$. Alice's only *safe* option is to choose $c'$ very close to $c$. But if the minimum distance of $C$ is sufficiently large, this forces $c' = c$.

# Quantum Cryptography

The following are problems with public key systems.

- They are based on the belief that some mathematical problem is hard, e.g. factorisation or computation of the discrete logarithm. This might not be true.
- As computers get faster, yesterday's securely encrypted message is easily read tomorrow.

The aim is to construct a key exchange scheme that is secure, conditional only on the laws of physics.

A classical bit is an element of $\{0,1\}$. A *quantum bit*, or *qubit*, is a linear combination $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $\alpha, \beta \in \mathbb{C}$, $|\alpha|^2 + |\beta|^2 = 1$. Measuring $|\psi\rangle$ gives $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$. After the measurement, the qubit collapses to the state observed, i.e. $|0\rangle$ or $|1\rangle$.

The basic idea is that Alice generates a sequence of qubits and sends them to Bob. By comparing notes afterwards, they can detect the presence of an eavesdropper.

$$\bigotimes \longrightarrow \boxtimes \longrightarrow \boxplus \longrightarrow$$

| light bulbs | polarity filter, angle $\vartheta$ to the vertical | vertical polarity filter |
|---|---|---|

Each photon passes through the second filter with probability $\cos^2\vartheta$. We identify $\mathbb{C}^2 = \{\alpha|0\rangle + \beta|1\rangle : \alpha, \beta \in \mathbb{C}\}$ with an inner product $(\alpha_1, \beta_1).(\alpha_2, \beta_2) = \alpha_1\bar{\alpha}_2 + \beta_1\bar{\beta}_2$. We can measure a qubit with respect to any orthonormal basis, e.g.

$$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$
$$|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

If $|\psi\rangle = \gamma|+\rangle + \delta|-\rangle$ then the observation gives $|+\rangle$ with probability $|\gamma|^2$ and $|-\rangle$ with probability $|\delta|^2$.

### BB84 (Bennet, Brassard, 1984)

- Alice sends Bob a stream of $(4+\delta)n$ qubits with randomly chosen polarisations $|0\rangle, |1\rangle, |+\rangle, |-\rangle$ with probability $\frac{1}{4}$.
- Bob measures the qubits, using either the first basis $|0\rangle, |1\rangle$ or the second basis $|+\rangle, |-\rangle$, deciding which at random.
- Afterwards, Alice announces which basis she used.
- Bob announces which bits he measured with the right bases. (There are about $(2 + \frac{\delta}{2})n$ of these.)

Now $A$ and $B$ share $2n$ bits. They compare $n$ of these bits and if they agree, use the other $n$ bits as their key.

**Remark.** An eavesdropper who could predict which basis Alice is using to send, or Bob uses to measure, could remain undetected. Otherwise, the eavesdropper will change about 25% of the $2n$ bits shared.

One problem is that noise has the same effect as an eavesdropper. Say $A$ and $B$ accept at most $t$ errors in the $n$ bits they compare, and assume at most $t$ errors in the other $n$ bits. Say $A$ has $x \in \mathbb{F}_2$, $B$ has $x + e \in \mathbb{F}_2$ with $\omega(e) \leq t$. We pick linear codes $C_2 \subset C_1 \subset \mathbb{F}_2^n$ of length $n$ where $C_1$ and $C_2^{\perp}$ are $t$-error correcting. $A$ chooses $c \in C_1$ at random and sends $x + c$ to $B$ using the clear channel. $B$ computes $(x + e) + (x + c) = c + e$ and recovers $c$ using the decoding rule for $C_1$.

To decrease the mutual information shared, $A$ and $B$ use as their key the coset $c + C_2$ in $C_1/C_2$.

This version of BB84 is provably secure conditional only on the laws of physics. A suitable choice of parameters can make both the probability that the scheme aborts and the mutual information simultaneously arbitrarily small.