

# Code Theory

Manuel Gijón Agudo

September 2017 - January 2018

# Índice

<b>1. Memoryless resources</b>	<b>3</b>
1.1. Sources and average word length . . . . .	3
1.2. Uniquely decodeable codes . . . . .	3
1.3. Optimal codes . . . . .	4
1.4. Extension of sources . . . . .	5
<b>2. Information and entropy</b>	<b>6</b>
2.1. Definitions . . . . .	6
2.2. Properties of the entropy function . . . . .	6
2.3. Shannon-Fano Code . . . . .	6
2.4. Product of sources . . . . .	7
2.5. Markov Chains . . . . .	7
2.6. Sources with memory . . . . .	8
<b>3. Information channels</b>	<b>9</b>
3.1. Channel matrix . . . . .	9
3.2. System Entropies and mutual information . . . . .	9
3.3. Extension of noiseless coding theorem to information channels . . . . .	10
3.4. Decision rules . . . . .	11
3.5. Improving reliability . . . . .	11
3.6. Rates of transmission and Hamming distance . . . . .	12
<b>4. Finite fields</b>	<b>14</b>
4.1. Basic definitions . . . . .	14
4.2. Properties of finite fields . . . . .	14
4.3. Factorization of polynomials . . . . .	15
<b>5. Block codes</b>	<b>17</b>
5.1. Minimum distance . . . . .	17
5.2. Bounds on block codes . . . . .	17
5.3. Asymptotically good codes . . . . .	18

<i>Code Theory</i>	2
<b>6. Linear codes</b>	<b>20</b>
6.1. Basics . . . . .	20
6.2. Syndrom decoding . . . . .	20
6.3. Dual code and Mc Williams identities . . . . .	20
6.4. The Griesmer bound . . . . .	20
<b>7. Cyclic codes</b>	<b>21</b>
7.1. Introduction . . . . .	21
7.2. Quadratic residue codes . . . . .	21
7.3. BCH Codes . . . . .	21
<b>8. Maximum distance separable codes</b>	<b>22</b>
8.1. Singleton bound . . . . .	22
8.2. Linear MDS codes . . . . .	22
<b>9. Alternant codes</b>	<b>23</b>
<b>10. Low density parity check codes</b>	<b>24</b>
10.1. Bipartite graphs with the expander property . . . . .	24
10.2. Low density parity check (LDPC) codes . . . . .	24
10.3. Belief propagation . . . . .	24
<b>11. P-adic codes</b>	<b>25</b>
11.1. P-adic numbers . . . . .	25
11.2. Polynomials over $\mathbb{Q}_p$ . . . . .	25

## 1. Memoryless resources

### 1.1. Sources and average word length

**Definition 1:** a **source** is a finite set  $\mathcal{S}$  together with a set of random variables  $(X_1, X_2, \dots)$  whose range is  $\mathcal{S}$ .

If  $P(X_n = \mathcal{S}_i)$  only depends on  $i$  and not on  $n$  then we say the source is **stationary** and if the  $X_n$  are independent then it's **memoryless**.

Insert example here

**Definition 2:** Let  $\mathcal{T}$  be a finite set called **alphabet**. A map  $\mathfrak{C} : \mathbb{S} \longrightarrow \mathbb{U}_{n \geq 1} T^n$  is called a **code**.

If  $|\mathcal{T}| = r$  then  $\mathfrak{C}$  is a  **$r$ -ary code**.

A code extends from  $\mathbb{S}$  to  $T \cup T^2 \cup \dots$  to  $\mathbb{S} \cup \mathbb{S}^2 \cup \dots$  to  $T \cup T^2 \cup \dots$  in obvious way.

insert example here

**Definition 3:** The **average word-length** of a code  $\mathfrak{C}$  is  $L(\mathfrak{C}) := \sum_{i=1}^n p_i l_i$  where  $l_i$  is the length of the image of the symbol of  $\mathbb{S}$ , which is emitted with probability  $p_i$ .

For now, we write  $\mathfrak{C}$  to be the image of  $\mathfrak{C}$ .

### 1.2. Uniquely decodeable codes

**Definition 4:** If for any sequences  $u_1 \dots u_n = v_1 \dots v_m$  in  $\mathfrak{C}$  implies  $m = n$  and  $u_i = v_i$  for  $i = 1, \dots, n$  then we say that  $\mathfrak{C}$  is **uniquely decodeable**.

insert example here

insert example here

insert example here

Let  $\mathfrak{C}_0 = \mathfrak{C}$ :

- $\mathfrak{C}_n := \{\omega \in T \cup T^2 \cup \dots \mid u\omega = v \text{ for some } u \in \mathfrak{C}_{n-1}, v \in \mathfrak{C} \text{ or } u\omega = v \text{ for some } u \in \mathfrak{C}, v \in \mathfrak{C}_{n-1}\}$
- $\mathfrak{C}_\infty := \bigcup_{k \geq 1} \mathfrak{C}_k$

Since everything is finite either  $\mathfrak{C}_m = \emptyset$  for some  $m$  and then  $\mathfrak{C}_n = \emptyset$  for  $n \geq m$  or it will be periodic and start repeating.

**Theorem 1:**  $\mathfrak{C}$  is uniquely decodeable  $\iff \mathfrak{C} \cap \mathfrak{C}_\infty = \emptyset$ .

*proof:* Insert proof here

insert example here

insert example here

insert example here

**Definition 5:** A code is a **prefix-code** if no codeword is prefix of another (ie.  $\mathfrak{C}_1 = \emptyset$ ).

A prefix code is uniquely decodeable.

**Theorem 2: (Kraft's inequality)**  $\exists r$ -ary prefix code with word lengths  $l_1, l_2, \dots, l_q \iff$

$$\sum_{i=1}^q r^{-l_i} \leq 1$$

*proof:* Insert proof here

insert example here

**Theorem 3: (McMillan's inequality)**  $\exists r$ -ary uniquely decodeable code with word lengths  $l_1, l_2, \dots, l_q \iff$

$$\sum_{i=1}^q r^{-l_i} \leq 1$$

*proof:* Insert proof here

### 1.3. Optimal codes

Let be  $\mathcal{S}$  a source with symbols  $s_1, \dots, s_q$  emitted with probabilities  $p_1, \dots, p_q$  and  $\mathfrak{C}$  is a code which encodes  $s_i$  with a codeword length  $l_i$ . Recall  $L(\mathfrak{C}) = \sum_{i=1}^q p_i l_i$ .

**Definition 6:** An **optimal code** for  $\mathcal{S}$  is an uniquely decodeable code  $\mathfrak{D}$  such that  $L(\mathfrak{C}) \geq L(\mathfrak{D})$  for all unique decodeable code  $\mathfrak{C}$ .

inset example here

insert example here

**Definition 7:** A code constructed in this way is called a **Huffman code**.

insert example here

Construct the  $r$ -arg Huffman code we sum together (at each step) the  $r$  smallest probabilities.

For this to work we need  $q \equiv 1(r-1)$ . Recall  $q$  is the number of symbols in the source. If not, then we add symbols with probabilities zero so that it is.

insert example here

**Lemma 1:** Every source  $\mathcal{S}$  has an optimal binary code  $\mathfrak{D}$  in which two of the longest codewords are **siblings**, ie.  $\exists x$  (a string) such that  $x_0, x_1 \in \mathfrak{D}$ .

*proof:* Insert proof here

**Theorem 4:** The Huffman code is an optimal code.

*proof:* Insert proof here

## 1.4. Extension of sources

Given a source  $\mathcal{S}$  we define  $\mathcal{S}^n$  the source with  $|\mathcal{S}|^n$  symbols, typically  $s_1, \dots, s_n$ , emitted with  $p_1, \dots, p_n$  probabilities.

insert example here

## 2. Information and entropy

### 2.1. Definitions

**Definition 1:** the **information** conveyed by a source is a function  $I : S \rightarrow [0, \infty)$  where  $S$  is a **source**<sup>1</sup> with the properties:

- $I(s_i)$  is a decreasing function of the propability  $p_i$ , with  $I(s_i) = 0$  if  $p_i = 1$ .
- $I(s_i s_j) = I(s_i) + I(s_j)$ , ie. the information gained by two symbols is the sum of the information obtained from each where the source has symbols  $s_1, \dots, s_q$  emitted with probabilities  $p_1, \dots, p_q$ .

**Lemma 1:**  $I(s_i) = -\log_r p_i$  for some  $r$ .

*proof:* Insert proof here

---

**Definition 2:** The  $r$ -ary **entropy**  $H_r(S)$  of a source  $S$  is the average information conveyed by  $S$ .

$$H_r(S) := - \sum_{i=1}^q p_i \log_r p_i$$

, by convection  $x \log_r x$  evaluated at 0 is 0.

Insert five examples

### 2.2. Properties of the entropy funcion

**Theorem 1:**  $H_r(S) \leq \log_r q$  with equality if and only iff  $S$  is the source where each symbol is emitted with probability  $1/q$ .

*proof:* Insert proof here

---

**Theorem 2:**  $H_r(S) \leq L(C)$  for unique decodeable code  $C$ .

*proof:* Insert proof here

---

### 2.3. Shannon-Fano Code

Let  $S$  be the source with symbols  $s_i$  and probabilities  $p_i$ . Let  $l_i := \lceil \log_r 1/p_i \rceil$ .

Then:  $\sum_{i=1}^q r^{-l_i} \leq \sum r^{-\log_r 1/p_i} = \sum p_i = 1$

---

<sup>1</sup>A **source** is a finite set  $S$  together with a sequence of random variables  $X_i$  whose range is  $S$

**Definition 3:** by Kraft exists a prefix code with word length  $l_1, l_2, \dots, l_1$ . This code is called **Shannon-Fano code**.

Inert example here

**Lemma 2:** For the Shannon-Fano code  $C$ :  $H_r(S) \leq L(C) < H_r(S) + 1$ .

*proof:* Insert proof here

---

## 2.4. Product of sources

Let  $S$  and  $T$  be two memoryless sources,  $S$  with symbols  $s_i$  and probabilities  $p_i$  and  $T$  with symbols  $t_j$  and probabilities  $q_j$ .

**Definition 4:** The **product source**  $S \times T$  is a source with symbols  $s_i t_j$  and probabilities  $p_i q_j$ .

**Theorem 3:**  $H_r(S \times T) = H_r(S) + H_r(T)$ .

*proof:* Insert proof here

---

**Corollary 1:**  $H_r(S^n) = nH_r(S)$ .

**Theorem 4: Noiseless Coding** The average word length  $L_n$  of an optimal code of  $S^n$  satisfies:

$$\frac{L_n}{n} \rightarrow H_r(S), n \rightarrow \infty$$

*proof:* Insert proof here

---

some examples

## 2.5. Markov Chains

**Definition 4:** A **Markov Chain** is a sequence of random variables where  $X_{n+1}$  depends only for  $X_n$ .

$$P(X_{n+1} = s_j | X_n = s_i) = p_{i,j}$$

This can be represented in a direct graph and also by a matrix  $P := (p)_{i,j}$ .

Suppose  $u_0$  is the vector which describes the initial distribution, ie. the  $i$ -th coordinate of  $u_0$  is probability we start at  $s_i$ . Probability of being in the  $i$ -th state after  $r$  steps is the  $i$ -th coordinate of  $u_0 P^r$ .

**Theorem 5:** if  $\exists r \in \mathbb{N}$  such that  $P^r$  has no zero entries, then  $u_0 P^r \rightarrow u$ , as  $n \rightarrow \infty$ .



**Definition 5:** This vector  $u$  is called the **stationary distribution**. It is normalised eigenvector of  $P^t$  with eigenvalue 1, ie.  $u_j = \sum_i p_{i,j} u_i$  and  $\sum_j u_j = 1$ .

**Definition 6:** If  $P$  is the matrix of a Markov Chain and  $\exists r$  such that  $P^r$  has non zero entries then we say that the Markov Chain is **regular**.

## 2.6. Sources with memory

Suppose  $S$  is a Markov Chain source with random variables  $X_1, X_2, \dots$  such that

$$P(X_{n+1} = s_j | X_n = s_j) = p_{i,j}$$

**Definition 7:**  $S$  is **not memoryless**, but it is stationary.

**Theorem 6:** suppose  $S$  is a regular Markov Chain source with stationary distribution  $u = (u_1, \dots, u_n)$ . Let  $S'$  be the stationary memoryless source with the same source elements as  $S$  (where  $s_i$  is emitted with probability  $w_i$ ). Then:

$$H_r(S) \leq H_r(S')$$

*proof:* Insert proof here

---

### 3. Information channels

#### 3.1. Channel matrix

Let  $\mathcal{A}$  be a stationary memoryless source with random variables  $X_1, X_2, \dots$  where  $P(X_n = a_i) = p_i$  for  $a_i \in \mathcal{A}$ .

Suppose we transmit  $\mathcal{A}$  through a channel  $\Gamma$ .

Let  $\mathcal{B}$  be a source with random variables  $Y_1, Y_2, \dots$  where  $P(Y_n = b_j) = q_j$

For  $b_j$  emerging from the channel:

$$\mathcal{A} \xrightarrow{\Gamma} \mathcal{B}$$

**Definition 1:** The **channel** is defined by a matrix  $(p_{ij})$  where  $p_{ij} = P(X_n = b_j | X_n = a_i)$  the probability we receive  $b_j$  given that  $a_i$  was sent,  $p_{ij}$ -**forward probabilities**. The **backwards probabilities** are  $q_{ij} = P(X_n = a_i | Y_n = b_j)$  and **joint probabilities**  $r_{ij} = P(X_n = a_i, Y_n = b_j)$

insert example here

inser example here (binary erasure channel)

#### 3.2. System Entropies and mutual information

**Definition 2:** We define the **input entropy** as:

$$H(\mathcal{A}) := - \sum_i p_i \log(p_i)$$

**Definition 3:** We define the **output entropy** as:

$$H(\mathcal{B}) := - \sum_j q_j \log(q_j)$$

We suppress the  $r$  (base) in the  $\log_r$  but it's always the same for every one.

Given that we have received  $b_j \in \mathcal{B}$ ,  $H(\mathcal{A} | Y_n = b_j) = - \sum_i q_{ij} \log(q_{ij})$ .

This is telling us the average information of  $\mathcal{A}$  knowing that  $Y_n = b_j$ .

If  $H(\mathcal{A} | Y_n = b_j) = 0$  then  $\exists m$  such that  $q_{ij} = 0$  for all  $i \neq m$  and  $q_{ij} = 1$  if  $i = m$ , ie.  $P(X_n = a_m | Y_n = b_j) = 1$ , ie. if we receive  $b_j$  then we know that  $a_m$  was sent.

If  $H(\mathcal{A} | Y_n = b_j) = H(\mathcal{A})$  then we learn nothing about  $\mathcal{A}$  when we receive  $b_j$  and this occurs when  $q_{ij} = P(X_n = a_i | Y_n = b_j) = P(X_n = a_i) = p_i$ .

**Definition 4:** Averaging over  $b_j \in \mathcal{B}$  we get the **condicional entropy**:

$$H(\mathcal{A} | \mathcal{B}) := - \sum_j P(Y_n = b_j) H(\mathcal{A} | Y_n = b_j) = - \sum_{i,j} q_j q_{ij} \log q_{ij}$$

Similary:

$$H(\mathcal{B}|\mathcal{A}) := - \sum_{i,j} p_i p_{ij} \log p_{ij}$$

**Definition 5:** The **joint entropy**:

$$H(\mathcal{A}, \mathcal{B}) := - \sum_{i,j} r_{ij} \log r_{ij}$$

insert example here

**Theorem 1:** For sources  $\mathcal{A}$  and  $\mathcal{B}$ :

$$H(\mathcal{A}, \mathcal{B}) = H(\mathcal{A}|\mathcal{B}) + H(\mathcal{B}) = H(\mathcal{B}|\mathcal{A}) + H(\mathcal{A})$$

*proof:* Insert proof here

**Definition 6:** We define the **mutual information** as the amount of information about  $\mathcal{A}$  we have learnt from  $\mathcal{B}$  and vice-versa:

$$I(\mathcal{A}, \mathcal{B}) := H(\mathcal{B}) - H(\mathcal{B}|\mathcal{A}) = H(\mathcal{A}) - H(\mathcal{A}|\mathcal{B})$$

If  $H(\mathcal{A}) = H(\mathcal{A}|\mathcal{B})$  then  $\mathcal{B}$  tells us nothing about  $\mathcal{A}$ , so  $I(\mathcal{A}, \mathcal{B}) = 0$ . This is an unrialiable channel and useless as a mean of communication.

If  $H(\mathcal{A}|\mathcal{B}) = 0$  then knowing  $\mathcal{B}$  we know everythin about  $\mathcal{A}$ , so  $I(\mathcal{A}, \mathcal{B}) = H(\mathcal{A})$ . This is the perfect situation because when we recive something, we know exactly what was sent.

insert example here

### 3.3. Extension of noiseless coding theorem to information channels

We have proved that given a source  $\mathcal{A}$  we can find an encoding of  $\mathcal{A}^n$  such that the average word lenght  $L_n$  satisfies  $\frac{L_n}{n} \rightarrow H(\mathcal{A})$ .

$\mathcal{A} \rightarrow \mathcal{B}$ , imagine we know  $\mathcal{B}$ .

**Lemma 1:**  $H(\mathcal{A}^n|\mathcal{B}^n) = nH(\mathcal{A}|\mathcal{B})$

*proof:* EXERCISE

**Theorem 2:** if  $\mathcal{B}$  is know then we can find encodings of  $\mathcal{A}^n$  such that the average word length  $L_n$  satisfies  $\frac{L_n}{n} \rightarrow H(\mathcal{A}|\mathcal{B})$ .

*proof:* Insert proof here

### 3.4. Decision rules

$$\mathcal{A} \xrightarrow{\Gamma} \mathcal{B}$$

Where  $\mathcal{A}$  is the **input**,  $\mathcal{B}$  is the **output** and  $\Gamma$  is the **channel**.

The channel is given by a matrix  $(p_{ij})$ ,  $p_{ij} = P(Y_n = b_j | X_n = a_i)$ . We defined  $r_{ij} = P(X_n = a_i | X_n = b_j)$ .

So if we recive  $b_j$  we should “decode”  $b_j$  as  $a_{j*}$  where  $r_{j*j} \geq r_{ij}$  for all  $i$ .

**Definition 7:** We would define our decision  $\Delta : \mathcal{B} \rightarrow \mathcal{A}$  as  $\Delta(b_j) := a_{j*}$ , this is called the **ideal observer rule**.

Howecer, most likely we only know  $p_{ij}$ ’s.

**Definition 8:** In **maximun likelihood decoding** we use the decision rule  $\Delta(b_j) := a_{j*}$ , where  $p_{j*j} \geq p_{ij}$  for all  $i$ .

**Definition 9:** The **average probability of a correct decoding** is:

$$P_{cor} := \sum_j q_j q_{j*j} - \sum_j r_{j*j}$$

Remind  $q_{ij} = P(X_n = a_i | Y_n = b_j)$ . Given that we received  $b_j$  if we dcode it as  $a_{j*}$  then the probability we have decoded correctly is  $P(X_n = a_{j*} | Y_n = b_j) = q_{j*j}$

### 3.5. Improving reliability

Suposse  $\Gamma$  is the binary symmetrical channel  $\begin{pmatrix} \phi & 1-\phi \\ 1-\phi & \phi \end{pmatrix}$  (and assume  $\phi > \frac{1}{2}$ ).

If we extends the source  $\mathcal{A} = \{0, 1\}$  to  $\{000, 001\}$  then the outpout source if  $\{000, 001, 010, 100, 110, 101, 011, 111\}$ . Now we have the channel matrix:

$$\begin{pmatrix} \phi^3 & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & (1-\phi)^3 \\ (1-\phi)^3 & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^2(1-\phi) & \phi^3 \end{pmatrix}$$

if we decode  $\Delta(000) = \Delta(001) = \Delta(010) = \Delta(100) = 0$  and  $\Delta(111) = \Delta(110) = \Delta(101) = \Delta(011) = 1$ .

effectively we have the channel:

$$\begin{pmatrix} \phi^3 + 3\phi^2(1-\phi) & 3\phi^2(1-\phi) + (1-\phi)^3 \\ 3\phi^2(1-\phi) + (1-\phi)^3 & \phi^3 + 3\phi^2(1-\phi) \end{pmatrix}$$

since  $\phi > 1 - \phi$  we have  $\phi^3 + 3\phi^2(1 - \phi) > \phi$ .

So we have proved the reliability of the channel, because  $P_{cor} = \sum_j r_{j*j} = p(\phi^3 + 3\phi^2(1 - \phi)) + (1 - p)(\phi^3 + 3\phi^2(1 - \phi)) = \phi^3 + 3\phi^2(1 - \phi)$ .

Observe if we do not extend the sorce  $P_{cor} = \phi$ .

### 3.6. Rates of transmission and Hamming distance

noindent Suppose  $\mathcal{A}$  is a source with  $r$  symbols. By extending the source, consider  $\mathcal{C}$  to be a subset of  $\mathcal{A}^n$ .

**Definition 10:** The **(transmission) rate of  $\mathcal{C}$**  is:

$$R := \frac{\log_r |\mathcal{C}|}{n}$$

By increasing  $n$  in the previous exercise we can make  $P_{cor} \rightarrow 1$ . However  $R \rightarrow 0$  since  $|\mathcal{C}| = \frac{\log_2 2}{n} \rightarrow 0$ .

**Definition 11:** The **capacity of a channel  $\Gamma$**  is:

$$\Lambda = \max_{\mathcal{A}, \mathcal{B}} I(\mathcal{A}, \mathcal{B})$$

Maximising over  $\mathcal{A}, \mathcal{B}$  means we can vary  $p_i$ 's and  $q_j$ 's.

Since  $\mathcal{C}$  is a subset of  $\mathcal{A}^n$  the rate tell us how many bits of information we can send in  $n$  bits (it is  $Rn$ ).

**Lemma 2:** The capacity of a binary symmetric channel  $\begin{pmatrix} \phi & 1-\phi \\ 1-\phi & \phi \end{pmatrix}$  is  $\Lambda = 1 + \phi \log_2 \phi + (1 - \phi) \log_2 (1 - \phi)$ .

*proof:* Insert proof here

**Definition 12:** For any  $u, v \in \mathcal{A}^n$ , the **Hamming distance** is  $d(u, v) :=$  number of coordinates where  $u$  and  $v$  differ.

**Lemma 3:** The Hamming distance satisfies the triangle inequality  $d(u, v) \leq d(u, w) + d(w, v)$

*proof:* Insert proof here

**Lemma 4:** For the binary symmetric channel, maximum likelihood decoding is  $\Delta(v) = u$ , where  $u$  is the closest element of  $\mathcal{C}$  with respect to the Hamming distance.

*proof:* Insert proof here

**Definition 13:** in general this decoding is called **nearest neighbour decoding**.

**Lemma 5:** For  $0 < \lambda < \frac{1}{2}$ :

$$\sum_{i=0}^{\lambda n} \binom{n}{i} \leq 2^{n(-\lambda \log(\lambda) - (1-\lambda) \log(1-\lambda))}$$

*proof:* Insert proof here

**Theorem 2: (Shannon)** Let  $\delta, \varepsilon > 0$ . For all sufficiently large  $n$  there is a code of length  $n$  and rate  $R$  satisfying  $\Lambda - \varepsilon < R < \Lambda$  together with a decision rule  $\Delta$  such that  $P_{cor} \rightarrow 1 - \delta$ .

*proof:* Insert proof here (ONLY FOR BINARY SYMMETRIC CHANNEL)

**Lemma 6:** For an input source  $\mathcal{A}$  and output source  $\mathcal{B}$  and decision rule  $\Delta(b_j) = a_{j*}$ .

$$H(\mathcal{A}|\mathcal{B}) \leq -P_{cor} \log(P_{cor}) - (1 - P_{cor}) \log(1 - P_{cor}) + (1 - P_{cor})(\log |\mathcal{C}| - 1)$$

where  $\mathcal{C}$  is the set of input source elements emitted with non zero probability.

**Theorem 3:** If  $\Lambda' > \Lambda$  and we fix the input probability distribution is uniform then there is no sequence of codes  $C_n$  of rate  $R$  satisfying  $\Lambda' - \varepsilon < R < \Lambda'$  such that  $P_{cor} \rightarrow 1$  as  $n \rightarrow \infty$ .

*proof:* Insert proof here

## 4. Finite fields

### 4.1. Basic definitions

**Definition 1:** A **field** is a commutable ring in which every non-zero element has a multiplicative inverse.

insert example here

inse example here

**Notation 1:** We denote as  $(f)$  with  $f \in \mathbb{F}_p[X]$ , the **ideal consisiting of all multiples of f**.

**Theorem 1:** if  $f$  is an irreducible polynomial of degree  $h$  then  $\mathbb{F}_p[X]/(f)$  is a finite field with  $p^h$  elements.

*proof:* Insert proof here

insert examples here

Exercise : construct a field wih 9 elements.

Let  $\mathbb{F}$  be a finite field. Let  $n$  minimal such that adding 1  $n$  times gives 0.

Since  $\overbrace{(1 + \dots + 1)}^n = \overbrace{(1 + \dots + 1)}^r \overbrace{(1 + \dots + 1)}^{n/r} = 0$  minimaliy implies that  $n = p$  is prime.

**Definition 2:** In this situation, we say that  $\mathbb{F}$  has **characteristic**  $p$ . If no such  $p$  exists then we say that  $\mathbb{F}$  has **characteristic zero**, in which case  $\mathbb{F} \supset \mathbb{Z}$  and so  $\mathbb{F} \supseteq \mathbb{Q}$ .

insert exercise here

### 4.2. Propierties of finite fields

**Theorem 2:** Let  $\mathbb{F}$  be a field with  $q$  elements. For all  $x \in \mathbb{F}$ .  $x^q = x$ .

*proof:* Insert proof here

The finite field with  $q$  elements is unique since it is the splitting field of the polynomial  $x^q - x \in \mathbb{F}_p[X]$ .

Considerer the map  $x \mapsto x^p$  in  $\mathbb{F}$  ( $q = p^h$ ).

$$(x + y)^p = \sum_{j=0}^p \binom{p}{j} x^j y^{p-j} = x^p + y^p$$

Observe that  $\binom{p}{j} = 0$  (modulo  $p$ ) for  $j = 1, \dots, p-1$ .

$$(x * y)^p = x^p y^p$$

So this map is partially an automorphism of  $\mathbb{F}_p$  since it preserves addition and multiplication.

**Definition 3:** This is called the **Frobenius automorphism**.

$$x \mapsto x^p \mapsto x^{p^2} \mapsto x^{p^3} \mapsto \dots \mapsto x^{p^{h-1}} \mapsto x$$

### 4.3. Factorization of polynomials

Let  $\mathbb{F}_p$  denote the unique finite field with  $q$  elements ( $q = p^h$ ).

**Lemma 1:** The polynomial  $x^{q-1} - 1$  factories into distinct linear factors in  $\mathbb{F}_q[X]$ .

*proof:* Insert proof here

**Lemma 2:** The polynomial  $x^q - 1$  factories into distinct irreducible factors whose degree divides  $h$ .

*proof:* Insert proof here

insert example here

insert example here

**Observation 1:** if  $q$  is odd  $x^{q-1} - 1 = (x^{\frac{q-1}{2}} - 1)(x^{\frac{q-1}{2}} + 1)$  the zeros of the first factor are on the non-zeros squares in  $\mathbb{F}_q$  and vice-versa ( $x = y^2$  then  $x^{\frac{q-1}{2}} = y^{q-1} = 1$ ).

**Observation 2:** if  $q^1 = q^r$  then  $x^n - 1 = (x^{n/q'} - 1)^{q'}$  so if we want to factorise  $x^n - 1$  in  $\mathbb{F}_p[x]$  we can assume  $(n, p) = 1$ .

To factorise  $x^n - 1$  in  $\mathbb{F}_q[X]$ , we find an extension field in  $\mathbb{F}_q$  which contains  $n$ -th roots of 1, ie. find  $h$  such that  $n$  divides  $q^h - 1$  since then  $x^{q^n-1} - 1$  is divisible by  $x^n - 1$ , ie.  $q^n = 1 \pmod{1}$ , ie.  $h$  is the multiplicative order of  $q$  in  $\mathbb{Z}/n\mathbb{Z}$ .

If we let  $\varepsilon$  be a primitive  $n$ -th root of 1 in  $\mathbb{F}_{q^n}$  then  $(x - \varepsilon)(x - \varepsilon^q)(x - \varepsilon^{q^2})\dots(x - \varepsilon^{q^{h-1}})$  is a polynomial whose coefficients are in  $\mathbb{F}_q$  since  $(x - \varepsilon)(x - \varepsilon^q)(x - \varepsilon^{q^2})\dots(x - \varepsilon^{q^h})$ .



insert example here

insert exercise here

insert example here

## 5. Block codes

### 5.1. Minimum distance

Let  $\mathcal{A}$  be a finite set (an alphabet).

**Definition 1:** A **block code**  $\mathfrak{C}$  of length  $n$  is a subset of  $\mathbb{A}^n$ .

**Definition 2:** The **minimum distance of**  $\mathfrak{C}$  is the minimum Hamming distance between any 2 codewords (elements of  $\mathfrak{C}$ ).

We are going to use nearest neighbour decoding so we want  $d$  as large as possible. We also can't  $|\mathfrak{C}|$  to be as large as possible.

**Lemma 1:** A block code of minimum distance  $d$  can correct up to  $\left\lceil \frac{d-1}{2} \right\rceil$  errors using nearest neighbour decoding.

*proof:* Insert proof here

insert example here

insert example here

**Definition 3:** Let  $\mathfrak{C}$  be a binary code of length  $n$ . The **extended code**  $\overline{\mathfrak{C}}$  is the code of length  $n+1$  defined by:

$$\overline{\mathfrak{C}} := \{(u_1, \dots, u_{n+1}) : u \in \mathfrak{C} \text{ where } u_{n+1} = u_1 + \dots + u_n \pmod{2}\}$$

**Theorem 1:** if the minimum distance  $d$  of a binary code is odd then the minimum distance of  $\widehat{\mathfrak{C}}$  is  $d+1$ .

*proof:* Insert proof here

### 5.2. Bounds on block codes

Let  $\mathcal{A}_r(n, d)$  denote the maximum  $|\mathfrak{C}|$ , such that exists a block code  $\mathfrak{C}$  of length  $n$ , minimum distance  $d$  over an alphabet with  $r$ -elements.

**Theorem 1: (Gilbert-Varshamov Bound)**

$$\mathcal{A}_r(n, d) \left( 1 + \binom{n}{1}(r-1) + \dots + \binom{n}{d}(r-1)^d \right) \geq r^n$$

*proof:* Insert proof here

**Recall 1:** we defined the binary entropy function as  $h(p) = -p \log p - (1-p) \log (1-p)$ .

**Corollary 1:** in the case  $r = 2$ :

$$\frac{1}{n} \log_2 \mathcal{A}_2(n, d) \geq 1 - h(\delta), \text{ where } \delta = \frac{d}{n}$$

**Definition 4:**  $\delta = \frac{d}{n}$  is called **relative minimum distance**.

*proof:* Insert proof here

**Theorem 2: (Sphere packing bound)**

$$\mathcal{A}_r(n, d) \left( 1 + \binom{n}{d}(r-1) + \dots + \binom{n}{t}(r-1)^t \right) \leq r^n \text{ where } t = \left\lceil \frac{d-1}{2} \right\rceil$$

*proof:* Insert proof here

**Definition 5:** A code meeting the Spheree-packing bound is called **perfect code**.

**Observation 1:** the parameteres  $(n, t, r)$  must be such that:

$$1 + \binom{n}{d}(r-1) + \dots + \binom{n}{t}(r-1)^t \text{ is a power of } r$$

insert example and exercise here

**Lemma 2: (Plotkin Lemma)** An  $r$ -ary code  $\mathfrak{C}$  of length  $n$  and minimum distance  $d$  satisfies  $|\mathfrak{C}| \left( d + \frac{n}{r} - n \right) \leq d$ .

*proof:* Insert proof here

insert exercise here

**Theorem 3: (Plotkin-Bound)** if  $\mathfrak{C}$  is a binary code of length  $n$ , minimum distance  $d < \frac{n}{2}$ . then:

$$|\mathfrak{C}| \leq d2^{n-2d+2}$$

*proof:* Insert proof here

### 5.3. Asymptotically good codes

We will construct and use short length codes which we can encode and decode quickly, this is very useful in many applications.

insert short examples here

However, in many cases we will have a lot of data and if we chop  $n$  bits into  $\frac{n}{n_0}$  chunks which we can send with  $P_{cor} = P$  close to 1.

$$P^{\frac{n}{n_0}} \longrightarrow 0$$

Let's suppose we have a binary code of length  $n$  and rate  $R$  (so  $|\mathfrak{C}| \approx 2^{nR}$ ).

In the proof of the Shannon's Theorem, we wed to the fact that the expected number of errors (using the binary symmetric channel) was  $(1-\phi)n$ , so if we are going to use the nearest neighbour decoding we need that  $d$  is also linear in  $n$  (as  $n$  gets very large), so we want  $\delta = \frac{d}{n} > 0$ .

**Definition 5:** We call the sequence codes of length  $n$ , where  $n \rightarrow \infty$  and  $\delta > 0$ .  $R > 0$ . **asymptotically good**.

inset exercise here

**Theorem 4: (Sphere packing bound)** Asymptotically (for  $n$  large):

$$R \leq 1 - h\left(\frac{\delta}{2}\right)$$

*proof:* Insert proof here

**Theorem 5: (Plotkin)** if  $\delta \leq \frac{1}{2}$  then  $R \leq 1 - 2\delta$ .

*proof:* Insert proof here

**Definition 6:** Let  $\mathcal{A}(n, d, \omega)$ , **The maximum size** of a binary code of length  $n$  with minimum distance  $d$  in which all the codewords have weight  $\omega$ .

(For any tuple  $v \in \mathcal{A}^n$  where  $0 \in \mathcal{A}$ , the **weight**  $wt(v) := \{ \text{number of non-zero coordinates that it has} \}$ ).

**Lemma 3:**

$$\mathcal{A}(n, d, \omega) \leq \frac{nd}{2\omega^2 - 2n\omega + dn}$$

*proof:* Insert proof here

**CONJECTURE:** there's no perfect constant (apart from the trivial bounds) weight codes.

**Theorem 6:** Let  $R$  be the rate of a sequence of asymptotically good binary codes if  $\delta < \frac{1}{2}$  then:

$$R < 1 - h\left(\frac{1}{2}(1 - \sqrt{1 - 2\delta})\right)$$

where  $h(p) = -p \log_2(p) - (1 - p) \log_2(1 - p)$

## 6. Linear codes

### 6.1. Basics

### 6.2. Syndrom decoding

### 6.3. Dual code and Mc Williams identities

### 6.4. The Griesmer bound

## 7. Cyclic codes

### 7.1. Introduction

### 7.2. Quadratic residue codes

### 7.3. BCH Codes

Decision problem, yes/no problem

## 8. Maximun distance separable codes

### 8.1. Syngleton bound

### 8.2. Linear MDS codes

## 9. Alternant codes



## 10. Low density parity check codes

### 10.1. Bipartite graphs with the expander property

### 10.2. Low density parity check (LDPC) codes

### 10.3. Belief propagation

## 11. P-adic codes

Breve comentario

### 11.1. P-adic numbers

### 11.2. Polynomials over $\mathbb{Q}_p$