Università degli Studi di Siena
Facoltà di Ingegneria

Lecture notes on

Information Theory and Coding

Mauro Barni
Benedetta Tondi

2012

Contents

Chapter 1

Measuring Information

Even if information theory is considered a branch of communication theory, it actually spans a wide number of disciplines including computer science, probability, statistics, economics, etc. The most basic questions treated by information theory are: how can 'information' measured? How can 'information' be transmitted?
From a communication theory perspective it is reasonable to assume that the information is carried out either by signals or by symbols. Shannon's sampling theory tells us that if the channel is bandlimited, in place of the signal we can consider its samples without any loss. Therefore, it makes sense to confine the information carriers to discrete sequences of symbols, unless differently stated. This is what we'll do throughout this lectures.

1.1 Modeling of an Information Source

It is common sense that the reception of a message about a certain event, for instance the result of an experiment, brings us *information.* Of course the information is received only if we do not know the content of the message in advance; this suggests that the concept of "information" is related to the ignorance about the result of the event. We gain information only because of our prior uncertainty about the event. We can say that the amount of a-priori uncertainty equals the amount of information delivered by the subsequent knowledge of the result of the experiment.
The first successful attempt to formalize the concept of information was made by Shannon, who is considered the father of Information Theory. In his paper "The mathematical Theory of Communication" (published in the Bell System Technical Journal, 1948) Shannon stated the inverse link between *information* and *probability.* Accordingly, the realization of an event gives

more information if it is less probable. For instance, the news that a football match between Barcelona and Siena has been won by Siena team carries much more information than the opposite.
Shannon's intuition suggests that information is related to randomness. As a consequence, information sources can be modeled by *random processes*, whose statistical properties depend on the nature of the information sources themselves. A discrete time information source X can then be mathematically modeled by a discrete-time random process $\{X_i\}$. The alphabet $\mathcal{X}$ over which the random variables X_i are defined can be either discrete ($|\mathcal{X}| < \infty$) or continuous when $\mathcal{X}$ corresponds to $\mathbb{R}$ or a subset of $\mathbb{R}$ ($|\mathcal{X}| = \infty$). The simplest model for describing an information source is the discrete memoryless source (DMS) model. In a DMS all the variables X_i are generated independently and according to the same distribution, i.i.d.. In this case, it is possible to represent a memoryless source through a unique random variable X.

1.2 Axiomatic definition of Entropy

The first effort that Shannon made was searching for a measure of the average information received when an event occurs. We now provide the entire proof procedure leading to Shannon's formula of the entropy for the DMS case.
Let X be a random variable describing a memoryless source with alphabet $\mathcal{X} = \{x_1, x_2, ..., x_n\}$. Concisely, let us call p_i the quantity $Pr\{X = x_i\}$. Then $p_i \geq 0 \ \forall i = 0, 1, ..., n$ and $\sum_i^n p_i = 1$. Let H be the (unknown) measure we look for. According to Shannon's intuition $H(X)$ must be a function of the probabilities according to which the symbols X are emitted, that is

$$H(X) = H_n(p_1, p_2, ..., p_n). \tag{1.1}$$

In addition, the function $H_n(p_1, p_2, ..., p_n)$ should have several intuitive properties. It is possible to formulate these properties as axioms from which we will deduce the specific form of the H function.
The four fundamental axioms are:

A.1 $H_2(\frac{1}{2}, \frac{1}{2}) = 1$ bit (*binary unit*).
This equality gives *the unit of measure of the information.*
It states that tossing a fair coin delivers 1 bit of information.

A.2 $H_2(p, 1-p)$ is a continuous function of p ($p \in [0, 1]$).
It expresses a natural requirement: small changes of probabilities of an experiment with two outcomes must result in small changes of the uncertainty of the experiment.

A.3 (*Permutation-invariance*)

$$H_n(\sigma(p_1, p_2, ..., p_n)) = H_n(p_1, p_2, ..., p_n), \tag{1.2}$$

for any permutation σ of the probabilities of the n symbols of the alphabet. The uncertainty of an experiment (i.e. the information delivered by its execution) does not depend on how the probabilities are assigned to the symbols.

A.4 (*Grouping property*)

$$H_n(p_1, p_2, ..., p_n) = H_{n-1}(p_1+p_2, p_3, ..., p_n)+(p_1+p_2)\cdot H_2\left(\frac{p_1}{p_1+p_2}, \frac{p_2}{p_1+p_2}\right), \tag{1.3}$$

where H_{n-1} is the measure of the information we receive from the experiment which considers the first two events grouped in a unique one, while the second terms gives the additional information regarding which of the two events occurred.

The above axioms reduce the number of possible candidate functions for H. Even more, it is possible to prove that they suffice to determine a unique function, as the next theorem asserts.
Let us extend the list of the axioms including another property. We will use it to prove the theorem. However, we point out that this is not a real axiom since it is deducible from the others. We will introduce it only to ease the proof.
We define $A(n) = H(\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n})$,

P.1 $A(n)$ is a *monotonically increasing* function of n.
This property is reasonable. If all the symbols are equally likely, the more they are, the more is the uncertainty about the result of the experiment.

Before stating and proving the theorem, we introduce some useful notations: let s_k be the partial sum of probabilities

$$\sum_{i=1}^{k} p_i = s_k, \tag{1.4}$$

and let $h(p)$ denote the entropy of the binary source $H_2(p, 1-p)$.

Theorem (*Entropy definition)***.**
There is only one function H_n which satisfies the four axioms listed above. Such a function has the following expression:

$$H_n(p_1, p_2, ..., p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i. \tag{1.5}$$

H_n is referred to as the *Entropy* of X.

Proof. The proof is organized in five steps.

1) By considering $A.4$ together with $A.3$ we deduce that we can group any two symbols, not only the first and the second one. We now want to extend the grouping property to a number k of symbols. We have:

$$\begin{aligned} H_n(p_1, p_2, ..., p_n) \quad &= \quad H_{n-1}(s_2, p_3, ..., p_n) + s_2 h\left(\frac{p_2}{s_2}\right) \\ &= \quad H_{n-2}(s_3, p_4, ..., p_n) + s_3 h\left(\frac{p_3}{s_3}\right) + s_2 h\left(\frac{p_2}{s_2}\right) = ... \\ ... &= \quad H_{n-k+1}(s_k, p_{k+1}, ..., p_n) + \sum_{i=2}^{k} s_i h\left(\frac{p_i}{s_i}\right). \end{aligned} \tag{1.6}$$

We would like to express the sum in (1.6) as a function of H_k. To this aim, we notice that starting from H_k and grouping the first $k-1$ symbols yields

$$\begin{aligned} H_k\left(\frac{p_1}{s_k}, ..., \frac{p_k}{s_k}\right) \quad &= \quad H_2\left(\frac{s_{k-1}}{s_k}, \frac{p_k}{s_k}\right) + \sum_{i=2}^{k-1} \frac{s_i}{s_k} h\left(\frac{p_i/s_k}{s_i/s_k}\right). \\ &= \quad \sum_{i=2}^{k} \frac{s_i}{s_k} h\left(\frac{p_i}{s_i}\right). \end{aligned} \tag{1.7}$$

By properly substituting the above equality in (1.6), we obtain the extension to k elements of the grouping property, that is

$$H_n(p_1, p_2, ..., p_n) = H_{n-k+1}(s_k, p_{k+1}, ..., p_n) + s_k H_k \left(\frac{p_1}{s_k}, ..., \frac{p_k}{s_k} \right). \tag{1.8}$$

2) Let us consider two integer values n and m and the function $A(n \cdot m)$. If we apply m times the extended grouping property we have just found (Point 1), each time to n elements in $A(n \cdot m)$, we obtain:

$$\begin{aligned} A(n \cdot m) &= H_{nm}\left(\frac{1}{nm}, ..., \frac{1}{nm}\right) \\ &= H_{nm-n+1}\left(\frac{1}{m}, \frac{1}{nm}, ..., \frac{1}{nm}\right) + \frac{1}{m} H_n \left(\frac{1}{n}, ..., \frac{1}{n}\right) \\ &\overset{(a)}{=} H_{nm-2n+2}\left(\frac{1}{m}, \frac{1}{m}, \frac{1}{nm}, ..., \frac{1}{nm}\right) + \frac{2}{m} A(n) = ... \\ ... &= H_m\left(\frac{1}{m}, ..., \frac{1}{m}\right) + A(n) \\ &= A(m) + A(n), \end{aligned} \tag{1.9}$$

where in (a) we implicity used axiom $A.3$.

3) From the previous point we deduce that

$$A(n^k) = k \cdot A(n). \tag{1.10}$$

We now consider the following property:

Property. The unique function which satisfies property (1.10) over all the integer values is the *logarithm function*. Then $A(n) = \log(n)$.

Proof. Let n be given. Then, for any arbitrary number r,

$$\exists k : \quad 2^k \leq n^r < 2^{k+1}. \tag{1.11}$$

By applying the (base-2) logarithm operator to each of the tree members, we obtain

$$k \leq r \log(n) < k + 1 \quad \rightarrow \quad \frac{k}{r} \leq \log(n) < \frac{k}{r} + \frac{1}{r}. \tag{1.12}$$

Hence, the distance between $\log(n)$ and k/r is at most $1/r$, i.e. $\left|\log(n) - \frac{k}{r}\right| < \frac{1}{r}$.

Similarly, we can apply the function A to the members of relation (1.11)[1]. By exploiting equality (1.10) and the fact that $A(2) = 1$ (Axiom 1), we get

$$\frac{k}{r} \leq A(n) < \frac{k}{r} + \frac{1}{r} \quad \text{or} \quad \left| A(n) - \frac{k}{r} \right| \leq \frac{1}{r}. \tag{1.13}$$

Therefore $|A(n) - \log(n)| \leq \frac{2}{r}$, which thanks to the arbitrariness of r concludes the proof. □

4) We are now able to show that the expression of the entropy in (1.5) holds for the binary case. Let us consider a binary source with $p = \frac{r}{s}$ for some positive values r and s (obviously, $r \leq s$).

$$\begin{aligned} A(s) &= \log(s) = H_s\left(\frac{1}{s}, ..., \frac{1}{s}\right) \\ &= H_{s-r+1}\left(\frac{r}{s}, \frac{1}{s}, ..., \frac{1}{s}\right) + \frac{r}{s} \cdot H_r\left(\frac{1}{r}, ..., \frac{1}{r}\right) \\ &= H_2\left(\frac{r}{s}, \frac{s-r}{s}\right) + \frac{s-r}{s} \cdot H_{s-r}\left(\frac{1}{s-r}, ..., \frac{1}{s-r}\right) + \frac{r}{s} \cdot A(r). \end{aligned} \tag{1.14}$$

From the last equality and thanks to point 3, we have

$$\log(s) = h\left(\frac{r}{s}\right) + \frac{s-r}{s} \cdot A(s-r) + \frac{r}{s} A(s). \tag{1.15}$$

By expliciting the term of the binary entropy we get

$$h\left(\frac{r}{s}\right) = \log(s) - \frac{s-r}{s} \cdot \log(s-r) - \frac{r}{s} \cdot \log(r). \tag{1.16}$$

Since $\log(s)$ can be written as $\frac{s-r}{s} \log(s) + \frac{r}{s} \log(s)$, we have

$$h\left(\frac{r}{s}\right) = -\frac{s-r}{s} \cdot \log\left(\frac{s-r}{s}\right) - \frac{r}{s} \cdot \log\left(\frac{r}{s}\right), \tag{1.17}$$

and then

$$h(p) = -(1-p) \cdot \log(1-p) - p \cdot \log(p) = \sum_{i=1}^{2} p_i \log p_i. \tag{1.18}$$

[1] Remember that $A(\cdot)$ is a monotonic function of its argument.

We have confined our derivation to rational probabilities. However, the following two considerations allow to extend our analysis to real numbers: the former is that rational numbers are dense in the real ones, the latter is that the h function is continuous $(A.2)$. Accordingly, for any irrational number p^* it is possible to construct a sequence of rational number p^n which tends to it for n tending to infinity. The corresponding sequence of the $h(p^n)$ values, thanks to $A.2$, has limit $h(p^*)$. This extends the proof to all $p \in \mathbb{R} \cap [0, 1]$.

5) As a last step, we extend the validity of the expression (1.5) to any value n. The proof is given by induction exploiting the relation for $n = 2$, already proved. Let us consider a generic value n and suppose that for $n - 1$ the expression holds. Then,

$$H_{n-1}(p_1, ..., p_{n-1}) = -\sum_{i=1}^{n-1} p_i \log p_i. \tag{1.19}$$

We want to show that the same expression holds for H_n.

$$\begin{aligned} H_n(p_1, ..., p_n) &= H_{n-1}(p_1 + p_2, p_3, ..., p_n) + (p_1 + p_2) \cdot h\left(\frac{p_1}{p_1 + p_2}\right) \\ &= -\sum_{i=3}^{n} p_i \log p_i - (p_1 + p_2) \cdot \log(p_1 + p_2) + \\ &\quad -\cancel{(p_1 + p_2)} \cdot \frac{p_1}{\cancel{(p_1 + p_2)}} \log\left(\frac{p_1}{p_1 + p_2}\right) + \\ &\quad -\cancel{(p_1 + p_2)} \cdot \frac{p_2}{\cancel{(p_1 + p_2)}} \log\left(\frac{p_2}{p_1 + p_2}\right) \\ &= -\sum_{i=3}^{n} p_i \log p_i - p_1 \log p_1 - p_2 \log p_2 \\ &= -\sum_{i=1}^{n} p_i \log p_i, \end{aligned} \tag{1.20}$$

which completes our proof. □

Information theory and statistical mechanics

The name *entropy* assigned to the quantity in (1.5) reminds the homonymous quantity defined in physics. Roughly speaking, from a microscopical point of view, Boltzmann defined the entropy S as the logarithm of the number of microstates Ω having an energy equal to E, i.e. $S = k \ln [\Omega(E)]$ [2], where k is a normalizing constant, by following a similar procedure to that adopted later by Shannon and described above. Boltzmann, who is one of the pioneers in statistical mechanics and thermodynamics, contributed in describing the entropy as a measure of the disorder of an individual, microscopic state of a physical system. Therefore, Shannon's concept of uncertainty is very similar to the disorder of a physical system. The analogies between information theory and statistical mechanics go far beyond this. However, we do not further dwell on this subject since is beyond the scope of these lectures.

1.3 Property of the Entropy

According to Shannon's definition, given a discrete random variable X with alphabet $\mathcal{X}$ and probability mass function $p(x) = Pr\{X = x\}$[3], $x \in \mathcal{X}$, the entropy $H(X)$ of the random variable X has the expression

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x), \tag{1.21}$$

where log is the base-2 logarithm. In (1.21) we use the convention that $0 \log 0 = 0$ which can be easily proved through de l'Hospital's rule. This is in agreement with the fact that adding zero probability terms does not change the value of the entropy.

Property. Let X be a DMS with alphabet $\mathcal{X}$ and $p(x)$ the corresponding pmf. Then,

$$H(X) \leq \log_2 |\mathcal{X}| \tag{1.22}$$

where the equality holds if and only if $p(x) = \frac{1}{|\mathcal{X}|}$ $\forall x \in \mathcal{X}$.

Proof. We exploit the relation $\ln z \geq 1 - 1/z$, where equality holds if and only if $z = 1$. Note that the logarithm involved in the relation is the base-e

[2] $\Omega(E)$ indicates the number of microstate having an energy equal to E.

[3] For convenience, we denote pmfs by $p(x)$ rather than by $p_X(x)$.

logarithm[4]. We have:

$$\begin{aligned}
\log_2|\mathcal{X}| - H(X) &= \log_2|\mathcal{X}| + \sum_{\mathcal{X}} p(x)\log_2(p(x)) \\
&= \sum_{\mathcal{X}} p(x)\left[\log_2|\mathcal{X}| + \log_2 p(x)\right] \\
&= \log_2 e \cdot \sum_{\mathcal{X}} p(x)\left[\ln|\mathcal{X}| + \ln p(x)\right] \\
&= \log_2 e \cdot \sum_{\mathcal{X}} p(x)\ln(|\mathcal{X}|p(x)) \\
&\overset{(a)}{\geq} \log_2 e \cdot \sum_{\mathcal{X}} p(x)\left(1 - \frac{1}{|\mathcal{X}|p(x)}\right) \\
&= \log_2 e \cdot \left(\sum_{\mathcal{X}} p(x) - \sum_{\mathcal{X}} \frac{1}{|\mathcal{X}|}\right) = 0. \qquad (1.23)
\end{aligned}$$

Hence,

$$\log_2|\mathcal{X}| \geq H(X), \qquad (1.24)$$

where the equality holds if and only if $p(x) = \frac{1}{|\mathcal{X}|}$, $\forall x$ (in which case (a) holds with the equality). □

From the above property, we argue that the uniform distribution for an information source is the one that gives rise to the maximum entropy. This fact provides new hints about the correspondence of information theory and statistical mechanics. In a physical system the condition of equally likely microstates is the configuration associated to the maximum possible disorder of the system and hence to its maximum entropy.

[4] Remember the relation $\log_2 z = \log_2 e \cdot \log_e z$ holding for logarithms with a different base, which will be useful in the following.

Chapter 2

Joint Entropy, Relative Entropy and Mutual Information

In Chapter 1 we defined the entropy of a random variable as the measure of the uncertainty of the random variable, or equivalently the measure of the amount of information required on the average to describe the value assumed by the random variable. In this chapter we introduce some related quantities.

2.1 Joint and Conditional Entropy

2.1.1 Joint Entropy

Given two discrete memoryless sources X and Y with alphabet $\mathcal{X}$ and $\mathcal{Y}$ respectively, we define the information obtained by observing the couple of random variables (X, Y). The extension of the entropy definition to a pair of random variables is called *joint entropy* and involves the joint distribution $p_{XY}(x, y)$, that is the statistical quantity describing the dependence between the variables.

Definition. The joint entropy $H(X, Y)$ of a pair of discrete random variables (X, Y) with joint distribution $p(x, y)$ is defined as

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x,y). \tag{2.1}$$

The *joint entropy* can also be seen as the entropy of the vector random variable $Z = (X, Y)$ whose alphabet is the cartesian product $\mathcal{X} \times \mathcal{Y}$.

The intuitive properties of a quantity describing the 'joint information' are captured by $H(X,Y)$; for example, it can be shown that if X and Y are two independent sources then

$$H(X,Y) = H(X) + H(Y). \tag{2.2}$$

Proof. We exploit the relation between the pmfs of independent sources, i.e. $p(x,y) = p(x)p(y)$, and proceeds with some simple algebra.

$$\begin{aligned} H(X,Y) &= -\sum_{xy} p(x,y) \log p(x,y) \\ &= -\sum_{xy} p(x,y) \log p(x)p(x) \\ &= -\sum_{x}\sum_{y} p(x,y) \log p(x) - \sum_{x}\sum_{y} p(x,y) \log p(y) \\ &= -\sum_{x} p(x) \log p(x) \sum_{y} p(y|x) - \sum_{y} p(y) \log p(y) \sum_{x} p(x|y) \\ &= H(X) + H(Y). \end{aligned} \tag{2.3}$$

□

The definition of *joint entropy* can be easily extended to m sources, X_1, X_2,....,X_m, as follows

$$H(X_1, X_2, ..., X_m) = -\sum_{x_1}\sum_{x_2}\cdots\sum_{x_n} p(x_1, x_2, ..., x_n) \log p(x_1, x_2, ..., x_m). \tag{2.4}$$

Accordingly, equation (2.4) represents the entropy of the joint random variable $(X_1, X_2, ..., X_m)$ taking values in the alphabet $\mathcal{X}_1 \times \mathcal{X}_2 .... \times \mathcal{X}_m$. Equation (2.2) can also be generalized to m independent sources, yielding

$$H(X_1, X_2, ..., X_m) = \sum_{i=1}^{m} H(X_i). \tag{2.5}$$

2.1.2 Conditional entropy

We now characterize the information received by observing a random variable X when we already know the value taken by another random variable Y. Reasonably, if the knowledge of Y gives us some information about X, the information carried by X will no more be $H(X)$.

Given a single realization $Y = y$, we define the entropy of the conditional distribution $p(x|y)$[1] as

$$H(X|Y=y) = -\sum_{x\in\mathcal{X}} p(x|y)\log p(x|y). \tag{2.6}$$

Definition. Given a pair of random variables (X,Y), the conditional entropy $H(X|Y)$ is defined as

$$H(X|Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x|y). \tag{2.7}$$

The conditional entropy can be expressed as the expected value of the entropies of the conditional distributions averaged over the conditioning random variable Y, i.e. $H(X|Y) = \sum_{y\in\mathcal{Y}} p(y)\cdot H(X|Y=y)$.

We now list some useful properties of the conditional entropy:

- X and Y independent $\Rightarrow H(X|Y) = H(X)$;

Proof. Suggestion: exploit the relation $p(x|y) = p(x)$ which holds for independent sources. □

- (*Chain Rule*)

$$H(X,Y) \overset{(a)}{=} H(X) + H(Y|X) \overset{(b)}{=} H(Y) + H(X|Y). \tag{2.8}$$

Equality (a) tells us that the information given by the pair of random variables (X,Y), i.e. $H(X,Y)$, is the same information we receive by considering the information carried by X $(H(X))$, plus the 'new' information provided by Y $(H(Y/X))$, that is the information that has not been given by the knowledge of X. Analogue considerations can be made for equality (b).

[1] In probability theory, $p(x|y)$ denotes the *conditional probability distribution* of X given Y, i.e. the probability distribution of X when Y is known to be a particular value.

Proof.

$$\begin{aligned}
H(X,Y) &= -\sum_{xy} p(x,y)\log p(x,y) \\
&= -\sum_{xy} p(x,y)\log p(y|x)p(x) \\
&= -\sum_{x}\sum_{y} p(x,y)\log p(y|x) - \sum_{x}\sum_{y} p(x,y)\log p(x) \\
&= H(Y|X) + H(X). \qquad (2.9)
\end{aligned}$$

The same holds for equality (b). □

- (*Generalized Chain Rule*)

By considering the case of m sources we get the *generalized chain rule*, which takes the form

$$\begin{aligned}
H(X_1, X_2, ..., X_m) &= \sum_{i=1}^{m} H(X_i|X_{i-1}, X_{i-2}, ..., X_1) \\
&= H(X_1) + H(X_2|X_1) + H(X_3|X_2, X_1) + ... \\
&\quad ... + H(X_m|X_{m-1}, ..., X_1). \qquad (2.10)
\end{aligned}$$

Proof. Suggestion: for $m = 2$ it has been proved above. The proof for a generic m follows by induction. □

By referring to (2.10) the meaning of the term ‘chain rule’ becomes clear: at each step in the chain we add only the new information brought by the next random variable, that is the novelty with respect to the information we already have.

- (*Conditioning reduces entropy*)

$$H(X|Y) \leq H(X). \qquad (2.11)$$

This relation asserts that the knowledge of Y can only reduce the uncertainty about X. Said differently, conditioning reduces the value of the entropy or at most leaves it unchanged if the two random variables are independent.

Proof.

$$\begin{aligned} H(X) - H(X|Y) &= \sum_x \sum_y p(x,y) \log \frac{p(x|y)}{p(x)} \\ &= \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \\ &= \log e \sum_x \sum_y p(x,y) \ln \frac{p(x,y)}{p(x)p(y)}. \qquad (2.12) \end{aligned}$$

By using the lower bound for the logarithm, $\ln z \geq 1 - \frac{1}{z}$, from (2.12) we get

$$\geq \log e \sum_x \sum_y p(x,y) \left(1 - \frac{p(x)p(y)}{p(x,y)}\right) = 0 \qquad (2.13)$$

where the equality holds if and only if X and Y are independent. □

Warning

Inequality (2.11) is not necessarily true if we refer to the entropy of a conditional distribution $p(X/y)$ for a given occurrence y, that is $H(X|y)$. The example below aims at clarifying this fact.

Let us consider the problem of determining the most likely winner of a football match. Suppose that the weather affects differently the performance of the two teams according to the values of the table 2.1; X is the random variable describing the outcome of the match $(1, \times, 2)$ and Y is the random variable describing the weather condition (rain, sun). By looking at the table of values we note that if it rains we are in great uncertainty about the outcome of the match, while if it's sunny we are almost sure that the winner of the match will be the first team. As a consequence, if we computed $H(X/Y = rain)$ we would find out that the obtained value is larger then $H(X)$. Because of the fact we are considering the conditioning to a particular event, this fact should not arouse any wonder since it is not in conflict with relation (2.11).

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

Proof. It directly follows from the chain rule and from relation (2.11). □

Y/X	1	×	2
rain	1/3	1/3	1/3
sun	9/10	1/10	0

Table 2.1: The table shows the probability of the various outcomes in the two possible weather conditions.

- (*Conditional Chain Rule*)

Given tree random variables X_1, X_2, X_3, the following relation holds:

$$H(X_1, X_2/X_3) = H(X_1/X_3) + H(X_2/X_1, X_3). \tag{2.14}$$

As usual it's easy to argue that the above relation can be generalized to any number m of sources.

- (*Mapping application*)

If we apply a deterministic function g to a given random variable X, i.e. a *deterministic processing*, the following relation holds:

$$H(g(X)) \leq H(X). \tag{2.15}$$

This means that we have less a priori uncertainty about $g(X)$ than about X; in other words, considering $g(X)$ in place of X causes a loss of information. The equality in (2.15) holds only if g is an *invertible function*.

Proof. By considering the joint entropy, we apply the chain rule in two possible ways, yielding

$$H(X, g(X)) = H(X) + H(g(X)/X) = H(X) \tag{2.16}$$

and

$$H(X, g(X)) = H(g(X)) + H(X/g(X)). \tag{2.17}$$

By equating the terms in (2.16) and (2.17) we obtain

$$H(g(X)) = H(X) - H(X/g(X)) \leq H(X). \tag{2.18}$$

The inequality holds since the term $H(X/g(X))$ is always greater then zero and reaches zero only if the function g is invertible, so that it's possible to recover X by applying g^{-1} to $g(X)$. If this is the case, knowing X or $g(X)$ is the same. □

2.2 Relative Entropy and Mutual Information

In this section we introduce two concepts related to the entropy: the *relative entropy* and the *mutual information.*

2.2.1 Relative Entropy

The relative entropy is a way to measure the distance between two pmfs $p(x)$ and $q(x)$ defined over a same alphabet.

Definition. The *relative entropy* or *Kullback-Leibler distance* or even *divergence* between two probability mass functions $p(x)$ and $q(x)$ is defined as

$$\mathcal{D}(p||q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}. \tag{2.19}$$

We used the conventions $0 \log 0 = 0$, $p \log \frac{p}{0} = \infty$ and $0 \log \frac{0}{0} = 0$. Despite the name "distance", the relative entropy is not a distance at all. In fact, although the *positivity* property is fulfilled (see below), the divergence is not a *symmetric* quantity and does not satisfy the *triangular inequality* (which are the other properties that a distance function must own).
According to a common interpretation, the relative entropy $D(p||q)$ is a measure of the inefficiency of assuming that the distribution is q when the true distribution is p. For instance, let us suppose we have a source X whose symbols are drawn according to an unknown distribution $p(x)$; if we knew another distribution $q(x)$ and decided to use it in order to construct a source coder, then $D(p||q)$ would represents the extra bits we have to pay for the encoding. This situation arises frequently in estimation problems, where p and q are respectively the true and estimated distribution of an observable set.

- (*Positivity*)

$$\mathcal{D}(p(x)||q(x)) \geq 0. \tag{2.20}$$

where the equality holds if and only if $p(x) = q(x)$.

Proof. Suggestion: apply the relation $\ln z \geq 1 - \frac{1}{z}$. □

As for the entropy, it is possible to define the conditional version of the relative entropy and prove the chain rule property.

Definition. The *conditional relative entropy* $\mathcal{D}(p(x|y)||q(x|y))$ is given by the average of the relative entropies between the conditional probability mass functions $p(x|Y=y)$ and $q(x|Y=y)$ over the probability mass function $p(y)$. Formally,

$$\begin{aligned}\mathcal{D}(p(x|y)||q(x|y)) &= \sum_{\mathcal{Y}} p(y) \sum_{\mathcal{X}} p(x|y) \log \frac{p(x|y)}{q(x|y)} \\ &= \sum_{\mathcal{X}} \sum_{\mathcal{Y}} p(x,y) \log \frac{p(x|y)}{q(x|y)}. \end{aligned} \tag{2.21}$$

- (*Chain rule for relative entropy*)

$$\mathcal{D}(p(x,y)||q(x,y)) = \mathcal{D}(p(y)||q(y)) + \mathcal{D}(p(x|y)||q(x|y)). \tag{2.22}$$

Proof. Suggestion: use the expression for the joint probability (conditional probability theorem) for both terms inside the argument of the logarithm and replace the logarithm with an appropriate sum of two logarithms. □

2.2.2 Mutual Information

We now introduce the concept of mutual information which allows to measure the amount of information that two variables have in common.

Definition. Consider two random variable X and Y with joint probability mass function $p(x,y)$ and marginal probability mass functions $p(x)$ and $p(y)$. The *mutual information* between the two random variables is obtained as the difference between the entropy of one random variable and the conditional entropy of the same random variable given the other, i.e.

$$I(X;Y) = H(X) - H(X/Y). \tag{2.23}$$

According to the intuition, the mutual information represents the reduction in the uncertainty of X due to the knowledge of Y.
Let us derive the explicit expression for $I(X;Y)$ as a function of the proba-

bilities by applying the definitions of entropy and conditional entropy:

$$\begin{aligned} I(X;Y) &= -\sum_{\mathcal{X}} p(x)\log p(x) + \sum_{\mathcal{X}}\sum_{\mathcal{Y}} p(x,y)\log p(x|y) \\ &\overset{(a)}{=} -\sum_{\mathcal{Y}}\sum_{\mathcal{X}} p(x,y)\log p(x) + \sum_{\mathcal{X}}\sum_{\mathcal{Y}} p(x,y)\log p(x|y) \\ &= \sum_{\mathcal{X}}\sum_{\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}, \end{aligned} \tag{2.24}$$

where in (a) we replaced $p(x)$ with $\sum_{\mathcal{Y}} p(x,y)$.

We now give some properties of the mutual information:

- (*Symmetry*)

$$I(X;Y) = I(Y;X). \tag{2.25}$$

This property tells that, as expected, the information that X has in common with Y is the same that Y has in common with X.

Proof. By referring to (2.24), from the commutative property of the product operator and the symmetry of $p(x,y)$ it's easy to deduce that it is possible to exchange X and Y in the definition of the mutual information. An alternative and interesting way to prove of the symmetry of the mutual information is by exploiting the relation between conditional and joint entropy. We have:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= H(X) - (H(X,Y) - H(Y)) \\ &= H(X) + H(Y) - (H(X) + H(Y|X)) \\ &= H(Y) - H(Y|X) = I(Y;X). \end{aligned} \tag{2.26}$$

□

- (*Positivity*)

$$I(X;Y) \geq 0 \tag{2.27}$$

Proof. The proof is exactly the same we used to show relation (2.11). However, there is another way to prove the positivity of $I(X;Y)$, that is through the application of the relation $\ln z \geq 1 - \frac{1}{z}$ to the expression in (2.24). Notice that the positivity of I has been already implicitly proved in Section 2.1.2 by proving the relation $H(X/Y) \leq H(X)$. □

- X, Y independent r.v. $\Leftrightarrow I(X;Y) = 0$;

The validity of the above assertion arises also from the following:

Observation.
The mutual information $I(X;Y)$ is the relative entropy between the joint distribution $p(x,y)$ and the product of the marginal distributions $p(x)$ and $p(y)$:

$$I(X;Y) = \mathcal{D}(p(x,y)||p(x)p(y)). \tag{2.28}$$

The more $p(x,y)$ differs from the product of the marginal distributions, the more the two variables are dependent and then the common information between them large.
Hence, the positivity of the mutual information directly follows from that of the relative entropy.

We now define the conditional mutual information as the reduction in the uncertainty of X due to the knowledge of Y when we know another random variable Z.

Definition. The *conditional mutual information* of the random variables X and Y given Z is defined as

$$\begin{aligned} I(X;Y|Z) &= H(X|Z) - H(X|Z,Y) \\ &= \sum_{\mathcal{X}}\sum_{\mathcal{Y}}\sum_{\mathcal{Z}} p(x,y,z)\log\frac{p(x,y|z)}{p(x|z)p(y|z)}. \end{aligned} \tag{2.29}$$

Notice that even in this case conditioning is referred to the average value of z.

- (*Chain rule for mutual information*)

$$I(X_1,X_2,....,X_m;Y) = \sum_{i=1}^{m} I(X_i;Y|X_{i-1},....,X_1). \tag{2.30}$$

We can indicate $I(X_1,X_2,....,X_m;Y)$ with the equivalent notation $I(\vec{X};Y)$ where the variable $\vec{X} = (X_1,X_2,....,X_m)$ takes values in $\mathcal{X}^m$. For $i = 1$ no conditioning is considered.

Proof.

$$\begin{aligned} I(X_1, X_2, ..., X_m; Y) &\overset{(a)}{=} H(X_1, X_2, ..., X_m) - H(X_1, ..., X_m|Y) \\ &\overset{(b)}{=} \sum_{i=1}^{m} H(X_i|X_{i-1}, ..., X_1) - \sum_{i=1}^{m} H(X_i|X_{i-1}, ..., X_1, Y) \\ &\overset{(c)}{=} \sum_{i=1}^{m} I(X_i; Y|X_{i-1}, ..., X_1), \end{aligned} \tag{2.31}$$

where in (a) we simply rewrote the mutual information as a function of the entropies, and in (b) we applied the chain rule and the conditional chain rule for the entropy. Substracting the terms of the two sums i by i yields the mutual information terms, (c).
An alternative way to prove (2.30) starts, as usual, from the explicit expression of the mutual information as a function of the probabilities and passes through some algebra operations.

□

Venn diagram
All the above relationships among the entropy and the related quantities ($H(X)$, $H(Y)$, $H(X,Y)$, $H(X/Y)$, $H(Y/X)$ and $I(X;Y)$) can be expressed in a Venn diagram. In a Venn diagram these quantities are visually represented as sets and their relationships are described as unions or intersections among these sets, as illustrated in Figure 2.1.

Exercise:
To practice with the quantities introduced so far, prove the following relations:

- $H(X, Y|Z) \geq H(X|Z)$;
- $I(X, Y; Z) \geq I(X; Z)$;
- $I(X; Z|Y) = I(Z; Y|X) - I(Z; Y) + I(X; Z)$.

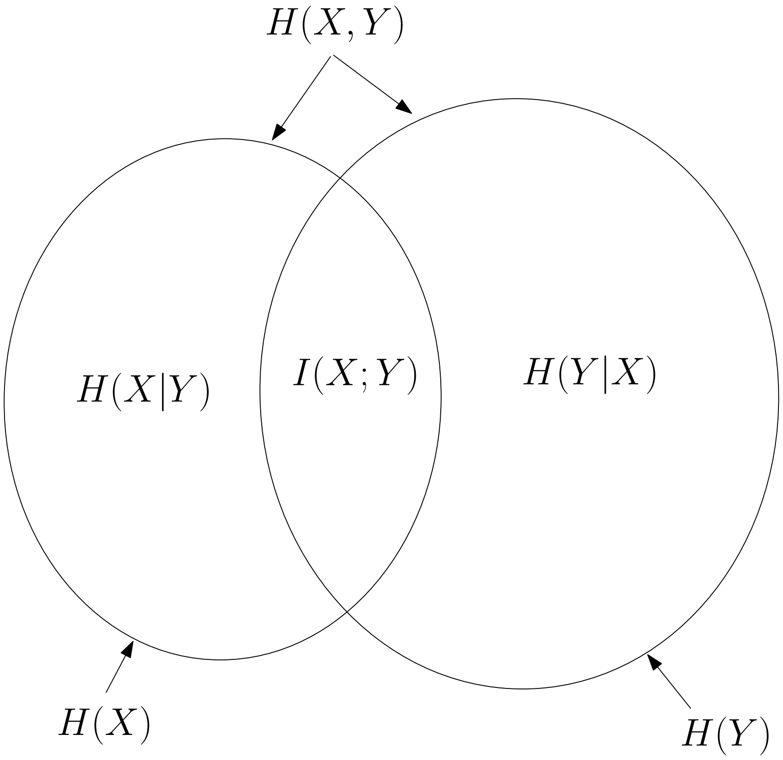


Figure 2.1: Venn diagram illustrating the relationship between entropy and mutual information.

Chapter 3

Sources with Memory

In the first two chapters we introduced the concept of information and some related measures confining our analysis to discrete memoryless sources. In this chapter we remove the memoryless assumption moving towards a more general definition of information.
Among the sources with memory, Markov sources play a major role. The rigorous definition of a Markov process will be given in Section 3.2.

3.1 Markov Chain (3 r.v.)

Let us consider the following configuration:

$$X \rightarrow Y \rightarrow Z. \tag{3.1}$$

Definition. Tree random variables X, Y and Z, form a *Markov chain* in that direction (denoted by $\rightarrow$) if

$$p(z|y,x) = p(z|y), \tag{3.2}$$

i.e., given Y, the knowledge of X (which precedes Y in the chain) does not change our knowledge about Z.

In a similar way, we can say that X, Y and Z form a Markov chain $X \rightarrow Y \rightarrow Z$ if the joint probability mass function can be written as

$$p(x,y,z) = p(z|y)p(y|x)p(x). \tag{3.3}$$

We now state some interesting properties of Markov chains.

Property (1).

$$X \to Y \to Z \quad \Leftrightarrow \quad p(x,z|y) = p(x|y)p(z|y), \tag{3.4}$$

that is the random variable X, Y and Z form a Markov chain with direction $\to$ *if and only if* X and Z are conditionally independent given Y.

Proof. We show first the validity of the direct implication, then that of the reverse one.

- M $\Rightarrow$ C.I. (*Markovity implies Conditional Independence*)

$$p(x,z|y) = \frac{p(x,y,z)}{p(y)} \overset{(a)}{=} \frac{p(z|y)p(y|x)p(x)}{p(y)} \overset{(b)}{=} p(z|y)p(x|y), \tag{3.5}$$

where in (a) we use the definition of Markov chain while equality (b) follows from Bayes' theorem;

- C.I. $\Rightarrow$ M. (*Conditional Independence implies Markovity*)

$$p(x,y,z) = p(x,z|y)p(y) \overset{(a)}{=} p(z|y)p(x|y)p(y) \overset{(b)}{=} p(z|y)p(y|x)p(x), \tag{3.6}$$

where the conditional independence between X and Z given Y yields equality (a), and equality (b) is again a consequence of Bayes' theorem.

□

Property (2).

$$X \to Y \to Z \quad \Rightarrow \quad Z \to Y \to X, \tag{3.7}$$

that is, if three random variable form a Markov chain in a direction, they also form a Markov chain in the inverse direction.

Proof. From Property (1) it's easy to argue that X and Y have an interchangeable rule; hence proving (3.7).

□

Observation.
If we have a deterministic function f, then

$$X \to Y \to f(Y). \tag{3.8}$$

In fact, since $f(\cdot)$ is a deterministic function of Y, if $f(Y)$ depends on X it is surely through Y; therefore, conditioning to Y makes X and $f(Y)$ independent.

Property (3). For a Markov chain the *Data Processing Inequality (DPI)* holds, that is

$$X \to Y \to Z \quad \Rightarrow \quad I(X;Z) \leq I(X;Y). \tag{3.9}$$

The DPI states that proceeding along the chain leads to a reduction in the information about the first random variable.

Proof. By exploiting the chain rule we expand the mutual information $I(X;Y,Z)$ in two different ways:

$$\begin{aligned} I(X;Y,Z) &= I(X;Z) + I(X;Y|Z) && (3.10)\\ &= I(X;Y) + I(X;Z|Y). && (3.11) \end{aligned}$$

By the properties of Markov chains we know that $I(X;Z|Y) = 0$. Then, from the positivity of the mutual information ($I(X;Y|Z) \geq 0$) the desired relation holds. Similarly, by reversing the direction of the chain (according to Property (2)), we can also prove that $I(X;Z) \leq I(Z;Y)$. □

Note: the data-processing inequality can be used to show that no clever manipulation of the data can increase the knowledge of the data and then improve the inferences that can be made from the data. With specific reference to the observation above, the DPI tells us that no deterministic processing of Y can increase the information that Y contains about X, i.e. $I(X;Y) \geq I(X;f(Y))$.

3.2 Characterization of Stochastic Processes

So far we have described a source of information by a random variable. Nevertheless, because of the introduction of memory in the source, this is no longer correct. Indeed, a source with memory has to be modeled as a *stochastic process*. For now, we limit our analysis to discrete time sources; accordingly, we refer to discrete-time processes. As to the symbols emitted by the sources, we still consider finite alphabets and then the processes we consider are also discrete-state.

The stochastic process describing a discrete memory source is then a sequence of random variables $X_1, X_2, ..., X_n$ which is often denoted by the notation $x(k,n)$, where the index k refers to the process sampling at a given instant, while the index n points out the process sampling in time for a given realization. Due to the presence of memory, the random variables representing the

output of the source at a given time instant are not necessarily identically distributed.
For simplicity, we shall use the notation $\mathbb{X}_n$ to represent the stochastic process omitting the dependence on k.
For mathematical tractability, we limit our analysis to stationary processes (stationary sources).

Definition. A stochastic process is said to be *stationary* if the joint distribution of any subset of the sequence of random variables is invariant with respect to shifts in the time index; that is

$$p_{X_1,...,X_n}(x_1,....,x_n) = p_{X_1+l,...,X_n+l}(x_1,....,x_n), \tag{3.12}$$

for every value n and every shift l and for all $x_1, x_2, ..., x_n \in \mathcal{X}$.

From a practical point of view, one may wonder whether such a model can actually describe a source in a real context. Well, it's possible to affirm that the above model represents a good approximation of some real processes, at least (if we consider them) on limited time intervals.

If a process is stationary each random variable X_i has the same distribution $p(x)$, whereby the entropy can still be defined and is the same for any i, i.e. $H(X_i) = H(X)$. However, as opposed to the memoryless case, the entropy no longer defines the information we receive by observing an output of the source when we know the previous outcomes. For sources with memory, in order to characterize such information, we need to introduce a new concept: the *Entropy Rate.*

3.2.1 Entropy Rate

We consider a sequence of n dependent random variables $X_1, X_2, ..., X_n$ describing a source with memory over an interval of n instants. If we observe n outputs we receive the amount of information $H(X_1, ..., X_n)$. It's clear that, due to the dependence among the variables, $H(X_1, ..., X_n) \neq nH(X)$. We want to determine how the entropy of the sequence grows with n. In this way, we would be able to get a definition of the growth rate of the entropy of the stochastic process i.e. its *entropy rate.*
Before stating the next theorem we first introduce two quantities that intuitively seem to be both reasonable definitions of the entropy rate.
We can define the average information carried by one of the n symbols emitted by the source as $H(X_1, ..., X_n)/n$. The question is: how large n should be in order to get a good estimation of the effective average? Clearly, it is

necessary to raise n so to take into account all the memory.
The quantity

$$\lim_{n\to\infty} \frac{H(X_1, ..., X_n)}{n}, \tag{3.13}$$

is the limit of the *per symbol entropy of the n random variables* as n tends to infinity and define, in literature, the *entropy of the stochastic process* $\{X_i\}$. An other quantity that seems to be a good definition for the amount of information received by observing the output is obtained as follows: take the conditional entropy of the last random variable given the past, and, to be sure to consider the entire memory of the source, take the limit, i.e.

$$\lim_{n\to\infty} H(X_n|X_{n-1}, ..., X_1). \tag{3.14}$$

It must be pointed out that, in general, the above limits may not exist. We now prove the important result that for stationary processes both limits (3.13) and (3.14) exist and assume the same value.

Theorem (*Entropy Rate)***.**
If $\mathbb{X}_n$ is a stationary source we have

$$\lim_{n\to\infty} \frac{H(X_1, ..., X_n)}{n} = \lim_{n\to\infty} H(X_n|X_{n-1}, ..., X_1), \tag{3.15}$$

which is defined *entropy rate* and denoted by $\mathcal{H}(\mathbb{X}_n)$.

Proof. We start by proving that the limit on the right-hand side of (3.15) exists. In fact

$$H(X_n|X_{n-1}, ..., X_1) \leq H(X_n|X_{n-1}, ..., X_2) = H(X_{n-1}|X_{n-2}, ..., X_1), \tag{3.16}$$

where the inequality follows from the fact that conditioning reduces the entropy, and the equality follows from the stationarity assumption. Relation (3.16) shows that $H(X_n|X_{n-i}, ..., X_1)$ is non-increasing in n. Since, in addition, for any n $H(X_n|X_{n-i}, ..., X_1)$ is a positive quantity, according to a well known result from calculus we can conclude that the limit in (3.14) exists and is finite.

We now prove that the average information $H(X_1, ..., X_n)/n$ has the same asymptotical limit value.
By the chain rule:

$$\frac{H(X_n, ..., X_1)}{n} = \sum_{i=1}^{n} \frac{H(X_i|X_{i-1}, ..., X_1)}{n}. \tag{3.17}$$

We indicate by a_n the quantity $H(X_n|X_{n-1},...,X_1)$ and by $\bar{a}$ the value $\lim_{n\to\infty} a_n$ which we know to exist and be finite; then, the average entropy is equivalent to $\sum_{i=1}^n a_i/n$. We can directly prove relation (3.15) by exploiting the following result from calculus:

$$a_n \to \bar{a} \quad \Rightarrow \quad \frac{\sum_{i=1}^n a_i}{n} \to \bar{a}. \tag{3.18}$$

Below, we give the formal proof.
Let us consider the absolute value of the difference between the mean value of a_i on n symbols and the value $\bar{a}$. According to the limit definition, we have to show that this quantity can be made arbitrarily small as $n \to \infty$. We can write:

$$\left|\frac{1}{n}\sum_{i=1}^n a_i - \bar{a}\right| = \frac{1}{n}\left|\sum_{i=1}^n (a_i - \bar{a})\right| \leq \frac{1}{n}\sum_{i=1}^n |a_i - \bar{a}|\,. \tag{3.19}$$

By the limit definition, $a_n \to \bar{a}$ means that

$$\forall \varepsilon > 0 \quad \exists N_\varepsilon : \forall n > N_\varepsilon \quad |a_n - \bar{a}| < \varepsilon. \tag{3.20}$$

Hence, going on from (3.19) we obtain

$$\frac{1}{n}\sum_{i=1}^n |a_i - \bar{a}| = \frac{1}{n}\sum_{i=1}^{N_\varepsilon} |a_i - \bar{a}| + \frac{1}{n}\sum_{i=N_\varepsilon+1}^{n} |a_i - \bar{a}|. \tag{3.21}$$

By looking at the first term of the sum we argue that its value is fixed (call it k) and finite while, thanks to the proper choice of N_ε, all the terms of the second summation are less then ε. Then we have

$$\left|\frac{1}{n}\sum_{i=1}^n a_i - \bar{a}\right| < \frac{k}{n} + \frac{n - N_\varepsilon + 1}{n}\cdot\varepsilon \quad \underset{n\to\infty}{\longrightarrow} \quad \varepsilon, \tag{3.22}$$

q.e.d.. □

Differently from the definition of the entropy, the definition of entropy rate is not so easy to handle for a generic source. The main difficulty resides in the *estimation of the joint distribution* of the source, which is needed for evaluating the joint or conditional entropy. All the more that, strictly speaking, the computation of the entropy rate requires that *n goes to infinity*, making the estimate of the joint distribution a prohibitive task. There are only some cases in which such estimation is possible: one of these is the case of Markov sources.

3.2.2 Markov Sources

In Section 3.1 we introduced the Markov chain for 3 random variables. We now give the general definition of a Markov process and discuss its principal features. For Markov sources we are able to evaluate the entropy of the process $\mathcal{H}(\mathbb{X}_n)$.

Definition (*Markov Chain)***.** A discrete stochastic process $\mathbb{X}_n$ is a *Markov chain* or *Markov process* if $\forall n$

$$p(x_n|x_{n-1},...,x_1) = p(x_n|x_{n-1}) \tag{3.23}$$

for all $x_1, x_2, ..., x_n \in \mathcal{X}^n$.

Then, in a Markov source the pmf at a given time instant n depends only on what happened in the previous instant $(n-1)$.

From (3.23) it follows that for a Markov source the joint probability mass function can be written as

$$p(x_1, x_2, ..., x_n) = p(x_n|x_{n-1})p(x_{n-1}|x_{n-2})...p(x_2|x_1)p(x_1). \tag{3.24}$$

For clarity of notation, we indicate through a_i a generic symbol of the source alphabet; then $\mathcal{X} = \{a_1, a_2, ..., a_m\}$ for some m.

Definition (*Time Invariant M.C.)***.** The Markov process is said to be *time invariant* (t.i.) if the conditional probability $p(X_n = a_j|X_{n-1} = a_i)$ does not depend on n; formally, for any n,

$$p(X_n = a_j|X_{n-1} = a_i) = p(X_2 = a_j|X_1 = a_i), \quad \forall a_i, a_j \in \mathcal{X}. \tag{3.25}$$

As a consequence, we can indicate the conditional probability of a time invariant Markov chain with the notation P_{ij}, without reference to the time index.
When $\mathbb{X}_n$ is a Markov chain, X_n is called the *state* at time n.
A t.i. Markov chain is completely characterized by the initial state, via the

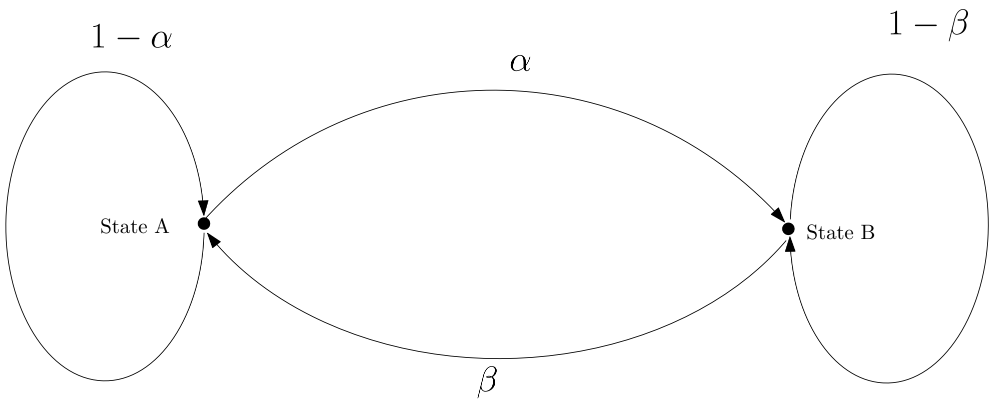


Figure 3.1: Two-state Markov chain.

probability vector $P^{(1)}$ [1], and the *probability transition matrix* $\mathbf{P} = \{P_{ij}\}$, $i, j = 1, 2, ..., |\mathcal{X}|$.
The probability vector at time n, which is

$$P^{(n)} = (p(X_n = a_1), p(X_n = a_2), ..., p(X_n = a_m))^T, \tag{3.26}$$

is recursively obtained by the probability vector at time $n-1$ as follows

$$P^{(n),T} = P^{(n-1),T} \cdot \mathbf{P}. \tag{3.27}$$

Observe that $P^{(n)} = P^{(n-1)}$ holds if $P^{(n)}$ is an eigenvector of $\mathbf{P}$ with eigenvalue 1 or if $\mathbf{P}$ is the identity matrix.
From now on, we assume that the Markov chain is time invariant unless otherwise stated.

Example (Two-state Markov chain).
Let us consider the simple example of a two state Markov chain shown in Figure 3.1 by means of a state diagram. The correspondent transition matrix is the following:

$$\mathbf{P} = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}.$$

We now introduce two interesting properties that a Markov chain may have.

[1] $P^{(1)}$ denotes the vector of the probabilities of the alphabet symbols at time instant $n = 1$, i.e. $P^{(1)} = (p(X_1 = a_1), p(X_1 = a_2), ..., p(X_1 = a_m))^T$.

Definition (*Irreducible M.C.*)**.** If it is possible to go, with positive probability, from any state of the chain to any other state in a finite number of steps, the Markov chain is said *irreducible*.

In particular, by referring to a same state i, we can define the period:

$$K_i = Lcf\{n : Pr\{X_n = i|X_0 = i\} > 0\}^2, \tag{3.28}$$

i.e. the largest common factor of the number of steps that starting from state i allow to come back to the state i itself (or equivalently, the largest common factor of the lengths of different paths from a state to itself).

Definition (*Aperiodic M.C.*)**.** If $K_i = 1 \; \forall i$, the (irreducible) Markov chain is said to be *aperiodic*.

Theorem (*Uniqueness of the stationary distribution*)**.**
If a finite state (t.i.) Markov chain is irreducible and aperiodic, there exists a *unique stationary distribution* $\Pi = \lim_{n\to\infty} P^{(n)}$ whatever the initial distribution $P^{(1)}$ is. This also means that:

$$\Pi^T = \Pi^T \cdot \mathbf{P}. \tag{3.29}$$

From the above theorem we deduce that the stationary distribution is so called because if the initial state distribution $P^{(1)}$ is itself Π the Markov chain forms a stationary process. If this is the case it is easy to evaluate the entropy rate by computing one of the two limits in (3.15).
In fact:

$$\begin{aligned} \mathcal{H}(\mathbb{X}_n) &= \lim_{n\to\infty} H(X_n|X_{n-1}, ..., X_1) \\ &\overset{(a)}{=} \lim_{n\to\infty} H(X_n|X_{n-1}) \\ &\overset{(b)}{=} \lim_{n\to\infty} H(X_2|X_1) \\ &= H(X_2|X_1), \end{aligned} \tag{3.30}$$

where (a) follows from the definition of Markov chain and (b) from the stationarity of the process. Hence, the quantity $H(X_2|X_1)$ is the *entropy rate of a stationary Markov chain*.
In the sequel we express the entropy rate of a stationary M.C. as a function of the quantities defining the Markov process, i.e. the initial state distribution $P^{(1)}$ and the transition matrix $\mathbf{P}$. Dealing with a stationary Markov chain

[2] *Lcf* is the abbreviation for largest common factor or greatest common divisor.

we know that $P^{(i)} = \Pi$ for all i, then we have

$$\begin{aligned} H(X_2|X_1) &= \sum_{i=1}^{|\mathcal{X}|} p(X_1 = a_i) H(X_2|X_1 = a_i) \\ &= \sum_{i=1}^{|\mathcal{X}|} P_i^{(1)} H(X_2|X_1 = a_i) \\ &= \sum_{i=1}^{|\mathcal{X}|} \Pi_i H(X_2|X_1 = a_i) \\ &= -\sum_{i=1}^{|\mathcal{X}|} \Pi_i \sum_{j=1}^{|\mathcal{X}|} P_{ij} \log P_{ij}, \qquad (3.31) \end{aligned}$$

where for a fixed i $p(X_2 = a_j|X_1 = a_i) = P_{ij}$ is the probability to pass from the state i to a state j, for $j = 1, ..., |\mathcal{X}|$. In general, $p(X_2|X_1 = a_i)$ corresponds to one element of the i-th row of the $\mathbf{P}$ matrix.

Going back to the example of the two state Markov chain we now can easily compute the entropy rate. In fact, by looking at the state diagram in Figure (3.1) it's easy to argue that the Markov chain is irreducible and aperiodic. Therefore we know that, for any starting distribution, the same stationary distribution is reached. The components of the vector Π are the stationary probabilities of the states A and B, i.e. Π_A and Π_B respectively. The stationary distribution can be found by solving the equation $\Pi^T = \Pi^T \cdot P$. Alternatively, we can obtain the stationary distribution by setting to zero the net probability flows across any cut in the state transition graph.
By imposing the balance at the cut to the state diagram in Figure (3.1) we have the following system with two unknowns

$$\begin{cases} \Pi_A \alpha = \Pi_B \beta, \\ \Pi_A + \Pi_B = 1, \end{cases}$$

where the second equality accounts for the fact that the sum of the probabilities must be one. The above system, once solved, leads to the following solution for the stationary distribution:

$$\Pi = \left(\frac{\beta}{\alpha + \beta}, \frac{\alpha}{\alpha + \beta} \right). \qquad (3.32)$$

We are now able to compute the entropy $\mathcal{H}(\mathbb{X}_n)$ from expression in (3.31).

Let us call $h(\alpha)$ the entropy of the binary source given the state A (i.e. $H(X_2|X_1 = A)$), that is the entropy of the distribution of the first row in $\mathbf{P}$. Similarly, we define $h(\beta)$ for the state B. The general expression for a two state Markov chain is

$$\mathcal{H}(\mathbb{X}_n) = \frac{\beta}{\alpha+\beta}h(\alpha) + \frac{\alpha}{\alpha+\beta}h(\beta). \tag{3.33}$$

Equation (3.33) tells us that in order to evaluate the entropy rate of the two state Markov chain it's sufficient to estimate the transition probabilities of the process once the initial phase ends.

Note: the stationarity of the process has not been required for the derivation. The entropy rate, in fact, is defined as a long term behavior and then is the same regardless of the initial state distribution. Hence, if the initial state distribution is $P^{(1)} \neq \Pi$ we can always skip the initial phase and consider the behavior of the process from a certain time onwards.

3.2.3 Behavior of the Entropy of a Markov Chain

We have already mentioned the relation between the entropy concept in information theory and the notion of entropy derived from thermodynamics. In this section we point out the similarity of a Markov chain with a *physical isolated system*. As in a Markov chain, even in a physical system knowing the present state makes the future of the system independent from the past: think for instance to the notion of position and velocities of gas particles. We now get more insight into the Markov chain $\mathbb{X}_n$ in order to show that the entropy rate $\mathcal{H}(\mathbb{X}_n)$ is nondecreasing, just as in thermodynamics.

Let $p^{(n)}$ and $q^{(n)}$ be two pmfs on the state space of a Markov chain at time n. According to the time invariance assumption these two distributions are obtained by starting from two different initial states $p^{(1)}$ and $q^{(1)}$. Let $p^{(n+1)}$ and $q^{(n+1)}$ be the corresponding distribution at time $n+1$, i.e. the evolution of the chain.

Property. The relative entropy $\mathcal{D}(p^{(n)}||q^{(n)})$ decreases with n; equivalently

$$\mathcal{D}(p^{(n+1)}||q^{(n+1)}) \leq \mathcal{D}(p^{(n)}||q^{(n)}) \quad \text{for any } n. \tag{3.34}$$

Proof. We use the expression $p^{(n+1,n)}$ $(q^{(n+1,n)})$ to indicate the joint probability distribution of the two discrete random variables, one representing the state at time n and the other representing the state at time $n+1$,

$$p^{(n+1,n)} = p_{X_{n+1},X_n}(x_{n+1}, x_n) \quad (q^{(n+1,n)} = q_{X_{n+1},X_n}(x_{n+1}, x_n)). \tag{3.35}$$

Similarly, by referring to the conditional distributions we have

$$p^{(n+1|n)} = p_{X_{n+1}|X_n}(x_{n+1}|x_n) \quad (q^{(n+1|n)} = q_{X_{n+1}|X_n}(x_{n+1}|x_n)). \tag{3.36}$$

According to the chain rule for relative entropy, we can write the following two expansions

$$\begin{aligned} \mathcal{D}(p^{(n+1,n)}||q^{(n+1,n)}) &= \mathcal{D}(p^{(n+1)}||q^{(n+1)}) + \mathcal{D}(p^{(n|n+1)}||q^{(n|n+1)}) \\ &= \mathcal{D}(p^{n}||q^{n}) + \mathcal{D}(p^{(n+1|n)}||q^{(n+1|n)}). \end{aligned} \tag{3.37}$$

It's easy to see that the term of $\mathcal{D}(p^{(n+1|n)}||q^{(n+1|n)})$ is zero, since in a Markov chain [3] the probability to pass from a state at time n to another state at time $n+1$ (the transitional probability) is the same whatever the probability vector is. Then, from the positivity of $\mathcal{D}$ equation (3.34) is proved. □

The above property asserts that in a Markov chain the K-L distance between the probability distributions tends to decrease as n increases.
We observe that equation (3.34) together with the positivity of $\mathcal{D}$ allows to say that the sequence of the relative entropies $\mathcal{D}(p^{(n)}||q^{(n)})$ admits limit as $n \to \infty$. However, we have no guarantee that the limit is zero. This is not surprising since we are working with a generic Markov chain and then the long term behavior of the chain may depend on the initial state (equivalently, the stationary distribution is not unique).

From the above property the following corollary holds.

Corollary. *Let* $p^{(n)}$ *be the state vector at time* n*; if we let* Π *be a stationary distribution, we have*

$\mathcal{D}(p^{(n)}||\Pi)$ *is a* ***monotonically non-increasing*** *sequence of* n,

thus implying that any state distribution gets closer and closer to each stationary distribution as time passes.

[3] Keep in mind that when we say "Markov chain" we implicity assume the time invariance of the chain.

As a consequence, if we also assume that the Markov chain is irreducible and aperiodic we know that the stationary distribution is unique and therefore the asymptotical limit of the sequence is zero, that is $\mathcal{D}(p^{(n)}||\Pi) \to 0$ as n grows.

The above corollary permits to state the following interesting property of Markov chains.

Property. If the stationary distribution is *uniform*, the entropy $H(X_n)$ increase as n grows

Proof. Due to the uniformity of the stationarity distribution, i.e. $\Pi_i = 1/|\mathcal{X}|$ for any $i = 1, 2, .., |\mathcal{X}|$, the relative entropy can be expressed as

$$\begin{aligned}\mathcal{D}(P^{(n)}||\Pi) = \sum_i P_i^{(n)} \log \frac{P_i^{(n)}}{\Pi_i} &= \sum_i P_i^{(n)} \log P_i^{(n)} + \sum_i P_i^{(n)} \log |\mathcal{X}| \\ &= \log |\mathcal{X}| - H(X_n). \qquad (3.38)\end{aligned}$$

Therefore, the monotonic decrease of the relative entropy as n grows implies the monotonic increase of the entropy $H(X_n)$. □

This last property has very close ties with statistical thermodynamics, which asserts that any closed system (remember that all the microstates are equally likely) evolves towards a state of maximum entropy or "disorder". If we further inspect equation (3.38), we can deduce something more than the relation $H(X_n) \to \log |\mathcal{X}|$. In fact, since $\mathcal{D}(P^{(n)}||\Pi)$ is monotonic non-increasing, the entropy growth with n is monotonic, meaning that the entropy does not swing (fluctuate).

We now briefly characterize the Markov processes having a uniform stationary distribution. Before, we give the following definition:

Definition. A probability transition matrix $P = \{P_{ij}\}$, where $P_{ij} = Pr\{X_{n+1} = j|X_n = i\}$, is called *doubly stochastic* if

$$\sum_i P_{ij} = 1, \quad j = 1, 2, ... \qquad (3.39)$$

and

$$\sum_j P_{ij} = 1, \quad i = 1, 2, ...^4 \qquad (3.40)$$

[4] This is always true. In any transition matrix the sum over the columns for a fixed row is 1.

We enunciate the following theorem, without giving the proof.

Theorem. A Markov chain admits a uniform stationary distribution *if and only if* the transition matrix $\mathbf{P}$ is doubly stochastic.

Chapter 4

Asymptotic Equipartition Property and Source Coding

In information theory, the asymptotic equipartition property (AEP) is a direct consequence of the weak law of large numbers defined in statistics. For simplicity, we confine the discussion to discrete memoryless sources.

4.1 A reminder of Statistics

According to the law of large number, given n independent and identically distributed (i.i.d.) random variables X_i, the sample mean $\frac{1}{n}\sum_{i=1}^{n} X_i$ is close to the expected value $E[X]$ for large values of n. The definition which is commonly adopted is that of the weak law of large numbers where with the term "weak" we refer to the *convergence in probability*. According to this type of convergence,

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \quad \underset{\text{in prob}}{\longrightarrow} \quad E[X] \tag{4.1}$$

means that:

$$\forall \varepsilon > 0, \quad Pr\{|\bar{X}_n - E[X]| > \varepsilon\} \longrightarrow 0 \quad \text{as} \quad n \to \infty. \tag{4.2}$$

As a reminder, the above equation is the same that, in statistics, proves the *consistence* of the point estimator $\bar{X}_n$, directly following from Tchebycheff inequality.
We point out that, although in (4.1)-(4.2) we considered the mean value $E[X]$, the convergence in probability of the sample values to the ensemble

ones can be defined also for the others statistics. Indeed, the law of large numbers rules the behavior of the relative frequencies k/n (where k is the number of successes in n trials) with respect to the probability p, that is it states that

$$Pr\left\{\left|\frac{k}{n} - p\right| > \varepsilon\right\} \longrightarrow 0 \quad \text{as} \quad n \to \infty. \tag{4.3}$$

Since the estimation of any statistic based on the samples depends on the behavior of the relative frequencies, the convergence in probability of the sample values to the ensemble ones can be derived from the law of large numbers.

Example (Repeated Trials).
This example illustrates the concept of the law of large numbers and at the same time introduces the concept of *typical sequence* formally defined later. Let the random variable $X \in \{0, 1\}$ model the toss of a fake coin (a smash). We set $0 =$ head, $1 =$ tail and assume that $p(1) = 0.9 = 1 - p$ and $p(0) = 0.1 = p$. We aim at showing that, when the numbers of tosses tends to infinity, the sequences drawn from X will have 90% of tails and 10% of heads with a probability arbitrarily close to 1.
Let $k = \alpha n$ be the number of 0's in the n-length sequence. Being a case of repeated trials, we have

$$Pr\{n(0) = k\} = \binom{n}{k} 0.1^k 0.9^{n-k}. \tag{4.4}$$

For any k, the probability of a sequence having k 0's, in the specific case $0.1^k 0.9^{n-k}$, tends to 0 as $n \to \infty$, while the binomial coefficient tends to ∞ [1], leading to an indeterminate form. By Stirling's formula[2] it's possible to prove that if we consider $k = pn = 0.1n$, $Pr\{n(0) = k\} \simeq 1$. All the corresponding sequences are referred to as *typical sequences*. As a consequence, all the other sequences (having a different number of 0's) occur with an approximately zero probability. We say that the sequences drawn from the sources have "the correct frequencies", where the term correct means that the relative frequencies coincide with the true probabilities.

[1] Strictly speaking, this is not true for the unitary sequence ($k = n$) which is only one and then has a zero probability.

[2] Stirling's formula gives an approximation for factorials: $n! \sim \left(\frac{n}{e}\right)^n \sqrt{2\pi n}$.

4.2 Asymptotic Equipartition Property

In this section we introduce one of the most important theorems of Information Theory which formally introduces the fundamental concept of *typicality.*
We consider a discrete memoryless source X with alphabet $\mathcal{X}$ and probability mass function $p(x)$. We use the notation X_i to indicate the random variable describing the outcome of the source at time i. According to the memoryless nature of the source, these variables are independent each other.
The asymptotic equipartition property (AEP) is formalized in the following theorem:

Theorem (*AEP).*
If $X_1, X_2, ...$ are i.i.d. $\sim p(x)$, then

$$-\frac{1}{n}\log p(X_1, X_2, ..., X_n) \underset{\text{in probability}}{\longrightarrow} H(X). \tag{4.5}$$

Proof.

$$\begin{aligned} -\frac{1}{n}\log p(X_1, X_2, ..., X_n) &= -\frac{1}{n}\log \prod_i^n p(X_i) \\ &= -\frac{1}{n}\sum_i^n \log p(X_i) \\ &= \frac{1}{n}\sum_i^n \log \frac{1}{p(X_i)}. \end{aligned} \tag{4.6}$$

By conveniently introducing the new random variables $Y_i = \log \frac{1}{p(X_i)}$, the above expression is just the sample mean of Y, i.e. $\frac{1}{n}\sum_i Y_i$. Therefore, by the law of large numbers we have that

$$-\frac{1}{n}\log p(X_1, ..., X_n) \to E[Y] \quad \text{in probability.} \tag{4.7}$$

Expliciting the expected value of Y yields

$$E[Y] = \sum_x p(x)\log\frac{1}{p(x)} = H(X). \tag{4.8}$$

□

The above theorem allows to give the definition of *typical set* and typical

sequence.
Let us rewrite relation (4.5) as follows

$$\left|-\frac{1}{n}\log p(X_1, X_2..., X_n) - H(X)\right| \to 0 \quad \text{in probability.} \tag{4.9}$$

According to the weak law of large numbers, equation (4.9) is equivalent to the following statement: $\forall\varepsilon > 0, \forall\delta > 0$

$$\exists N : \forall n > N \quad Pr\left\{\left|-\frac{1}{n}\log p(X_1, X_2, ..., X_n) - H(X)\right| > \varepsilon\right\} < \delta. \tag{4.10}$$

Hence, with high probability, the sequence $X_1, X_2, ..., X_n$ satisfies the relation

$$H(X) - \varepsilon \leq -\frac{1}{n}\log p(X_1, X_2, ..., X_n) \leq H(X) + \varepsilon, \tag{4.11}$$

which corresponds to the following lower and upper bound for the probability:

$$2^{-n(H(X)+\varepsilon)} \leq p(X_1, X_2, ..., X_n) \leq 2^{-n(H(X)-\varepsilon)}. \tag{4.12}$$

Definition. The *typical set* $A_\varepsilon^{(n)}$ with respect to $p(x)$ is the set of sequences $x^n = (x_1, x_2, ..., x_n) \in \mathcal{X}^n$ for which

$$2^{-n(H(X)+\varepsilon)} \leq p(x_1, x_2..., x_n) \leq 2^{-n(H(X)-\varepsilon)}. \tag{4.13}$$

We now give some informal insights into the properties of the *typical set*, from which it is already possible to grasp the key ideas behind Shannon's source coding theorem.
By the above definition and according to the law of large numbers we can argue that

$$Pr\{X^n \in A_\varepsilon^{(n)}\} \to 1 \quad \text{as } n \to \infty. \tag{4.14}$$

Then, the probability of any observed sequence will be *almost surely* close to $2^{-nH(X)}$. A noticeable consequence is that the sequences inside the typical set, i.e. the so called *typical sequences*, are **equiprobable**. Besides, since a sequence lying outside the typical set will almost never occur for large n, the number of typical sequences k can be roughly estimated as follows:

$$2^{-nH(X)} \cdot k \cong 1 \quad \Rightarrow \quad k \cong 2^{nH(x)}.$$

It is easy to understand that, in the coding operation, these are the sequences that really matter. For instance, if we consider binary sources, the above relation states that $nH(X)$ bits suffice on the average to describe n

binary random variables, leading to a considerable bit saving (remember that $H(X) < 1!$). These considerations concerning the typical set are the essence of Shannon's source coding theorem and will be made rigorous in the following section. The theorem below gives a rigorous formalization to the properties of $A_\varepsilon^{(n)}$, and is a direct consequence of the AEP.

Theorem (*Typical Set)*.
Let $X \sim p(x)$ be a DM source and $A_\varepsilon^{(n)}$ the correspondent typical set as defined above:

1. $\forall\delta, \forall\varepsilon, n$ large, $\quad \Pr\{A_\varepsilon^{(n)}\} \geq 1 - \delta$;
2. $\forall\varepsilon, \quad |A_\varepsilon^{(n)}| \leq 2^{n(H(X)+\varepsilon)} \quad \forall n$;
3. $\quad \forall\delta, \forall\varepsilon, n$ large, $\quad |A_\varepsilon^{(n)}| \geq (1-\delta)2^{n(H(X)-\varepsilon)}$.

Proof.
1. It directly follows from equation (4.10) which can also be written as follows: $\forall\varepsilon > 0, \forall\delta > 0$

$$\exists N : \forall n > N \quad Pr\left\{\left|-\frac{1}{n}\log p(X_1, ..., X_n) - H(X)\right| < \varepsilon\right\} > 1 - \delta. \tag{4.15}$$

Since the expression in curly braces defines the typical set, equation (4.15) proves point 1.

2.

$$\begin{aligned} 1 &= \sum_{x^n \in \mathcal{X}^n} p(x^n) \\ &\geq \sum_{x^n \in A_\varepsilon^{(n)}} p(x^n) \\ &\overset{(a)}{\geq} \sum_{x^n \in A_\varepsilon^{(n)}} 2^{-n(H(X)+\varepsilon)} \\ &= \left|A_\varepsilon^{(n)}\right| \cdot 2^{-n(H(X)+\varepsilon)} \end{aligned} \tag{4.16}$$

where (a) follows by the definition of typical set.
Notice that the proof of point 2 does not involve the bounds on the probability of the observed sequence and then holds for any n.

3. $\forall\delta > 0$ and for large n, from point 1. we have

$$\begin{aligned} 1-\delta &\leq Pr\{A_{\varepsilon}^{(n)}\} \\ &= \sum_{x_n \in A_{\varepsilon}^{(n)}} p\{x^n\} \\ &\leq \sum_{x_n \in A_{\varepsilon}^{(n)}} 2^{-n(H(X)-\varepsilon)} \\ &= \left|A_{\varepsilon}^{(n)}\right| \cdot 2^{-n(H(X)-\varepsilon)} \end{aligned} \tag{4.17}$$

□

4.3 Source Coding

In this chapter we state and prove Shannon's source coding theorem for the discrete memoryless case. We also discuss the extension of Shannon's theorem to the source with memory.

4.3.1 Memoryless Source Coding

The celebrated Shannon's source coding theorem, also known as *noiseless source coding theorem*, refers to the case of discrete memoryless sources. The theorem consists of two distinct parts: the *direct* theorem and the *converse* theorem.

Before stating the theorem, we need to give the definition of code and extended code. Let X be the source we want to compress with alphabet $\mathcal{X}$ and pmf P_X. Working on symbols, we define a coding as a mapping procedure from the source alphabet $\mathcal{X}$ to a code alphabet $\mathcal{C}$. Due to its use in computer science, we consider a binary code alphabet.

Definition (Binary code)**.** A binary code C associates to each source symbol x a string of bits[3], i.e. is a mapping $C : \mathcal{X} \to \{0,1\}^*$. For each x, $C(x)$ denote the associated codeword.

Definition (Expected length)**.** The expected length L of a code C for a random variable X with probability mass function $p(x)$ is given by

$$L = \sum_{x\in\mathcal{X}} p(x)l(x), \tag{4.18}$$

[3] A string of bits is an element in $\{0,1\}^*$, i.e. the set of all the binary strings.

where $l(x)$ is the length of the codeword associated with x.

We now define a property that a code should have.

Property (Non singular code)**.** A code is said *invertible* or *nonsingular* if each symbol is mapped into a different string, that is

$$a_i \neq a_j \Rightarrow C(a_i) \neq C(a_j), \quad a_i, a_j \in \mathcal{X}. \tag{4.19}$$

Since we transmit and store sequences of symbols, the above property does not guarantee the unambiguous description of the sequences and then their correct decodability. We need to define a further property which passes through the following definition:

Definition (Extended Code)**.** The *n-th extension* C^* of a code C is the mapping from n-length strings of elements in $\mathcal{X}$ to n-length strings of $\{0,1\}$, that is $C^* : \mathcal{X}^n \rightarrow \{0,1\}^*$. C^* is defined as the concatenation of the codewords of C:

$$C^*(x_1 x_2 ... x_n) = C(x_1)C(x_2)\cdots C(x_n). \tag{4.20}$$

Property (Uniquely decodable code)**.** A code is *uniquely decodable* if its extension is nonsingular.

We now start by enunciating the direct part of Shannon's source coding theorem.

Theorem (*Shannon's Source Coding: direct)***.**
Let X be a discrete memoryless source, with alphabet $\mathcal{X}$, whose symbols are drawn according to a probability mass function $p(x)$, and let x^n be a n length sequence of symbols drawn from the source. Then, $\forall \varepsilon > 0$ and sufficiently large n, $\exists\ C(x^n)$ invertible s.t.

$$\frac{L}{n} \leq H(X) + \varepsilon, \tag{4.21}$$

where L denotes the average length of the codewords, i.e. $E[l(c(x^n))]$, and L/n is the code rate, i.e. the average number of bits per symbol.

Proof. The proof comes out directly from the AEP theorem and Typical set theorem.
We search for a code having a rate which satisfies relation (4.21). In order to prove the theorem it's sufficient to find one such a code.
Let us construct a code giving a short description of the source. We divide

all sequences in $\mathcal{X}^n$ into two sets: the typical set $A_\varepsilon^{(n)}$ and the complementary set $A_\varepsilon^{(n),c}$.
As to $A_\varepsilon^{(n)}$, we know from the AEP theorem that the sequences x^n belonging to it are equiprobable and then we can use the same codeword length $l(x^n)$ $(l(C(x^n)))$ for each of them. We represent each typical sequence by giving the index of the sequence in the set. Since there are at most $2^{n(H(X)+\varepsilon)}$ sequences in $A_\varepsilon^{(n)}$, the indexing requires no more then $n(H+\varepsilon)+1$ bits, where the extra bit is necessary because $n(H+\varepsilon)$ may not be an integer. Spending another bit 0 as a flag, so to make uniquely decodable the code, the total length of bits is at most $n(H+\varepsilon)+2$. To sum up,

$$x^n \in A_\varepsilon^{(n)} \quad \Rightarrow \quad l(x^n) \leq n(H+\varepsilon)+2. \tag{4.22}$$

We stress that the order of indexing is not important, since it does not affect the average length.
As to the encoding of the non-typical sequences, the Shannon's idea is "to squander". Since the AEP theorem asserts that, as n tends to infinity, the sequences in the non-typical set $A_\varepsilon^{(n),c}$ will never occur, it's not necessary to look for a short description. Specifically, Shannon suggested indexing each sequence in $A_\varepsilon^{(n),c}$ by using no more than $n\log|\mathcal{X}|+1$ bits (as before, the additional bit takes into account the fact that $n\log|\mathcal{X}|$ may not be integer). Observe that $n\log|\mathcal{X}|$ bit would suffice to describe all the sequences ($|A_\varepsilon^{(n)}|$ + $|A_\varepsilon^{(n),c}| = |\mathcal{X}|^n$). Therefore, by using such coding, we waste a lot of bits (surprisingly, this is good enough to yield an efficient encoding). Prefixing the indices by 1, we have

$$x^n \in A_\varepsilon^{(n)c} \quad \Rightarrow \quad l(x^n) \leq n\log|\mathcal{X}|+2. \tag{4.23}$$

The description of the source provided by the above code is depicted in Figure 4.1. By using this code we now prove the theorem.
The code is obviously invertible. We now compute the average length of the codeword:

$$\begin{aligned} E[l(x^n)] &= \sum_{x^n} p(x^n)l(x^n) \\ &= \sum_{x^n\in A_\varepsilon^{(n)}} p(x^n)l(x^n) + \sum_{x^n\in A_\varepsilon^{(n),c}} p(x^n)l(x^n) \\ &\leq (n(H(X)+\varepsilon)+2)\cdot \sum_{x^n\in A_\varepsilon^{(n)}} p(x^n) + (n\log|\mathcal{X}|+2)\cdot \sum_{x^n\in A_\varepsilon^{(n),c}} p(x^n). \end{aligned} \tag{4.24}$$

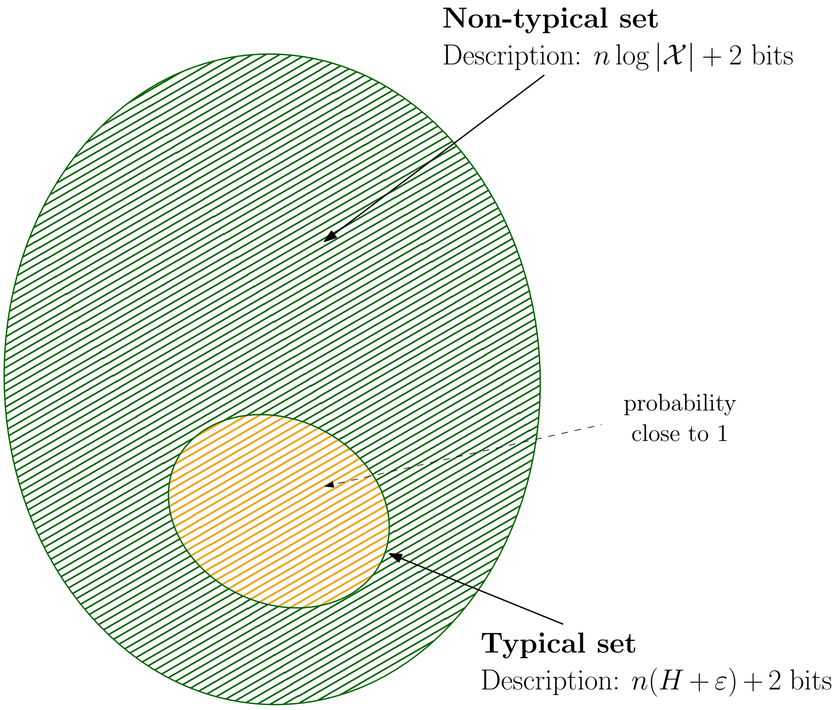

Non-typical set
Description: $n \log |\mathcal{X}| + 2$ bits
probability
close to 1
Typical set
Description: $n(H + \varepsilon) + 2$ bits

Figure 4.1: Source code using the typical set.

For all positive value δ, if n is sufficiently large, $Pr\{A_\varepsilon^{(n)}\} \geq 1 - \delta$; then expression (4.24) is upper bounded as follows:

$$\leq \quad (n(H(X) + \varepsilon) + 2) + \delta \cdot (n \log |\mathcal{X}| + 2). \tag{4.25}$$

Then,

$$\begin{aligned} \frac{L}{n} &\leq H(X) + \varepsilon + \frac{2}{n} + \delta \frac{2}{n} + \delta \log |\mathcal{X}| \\ &= H(X) + \varepsilon', \end{aligned} \tag{4.26}$$

where $\varepsilon' = \varepsilon + \frac{2}{n} + \delta \frac{2}{n} + \delta \log |\mathcal{X}|$ can be made arbitrarily small for an appropriate choice of δ and n.
It's clear that,since the non-typical sequences will almost never occur for large n, the lengths of their codewords have a negligible impact on the average codeword length.

□

The above theorem states that the code rate L/n can get arbitrarily close to the entropy of the source. Nevertheless, in order to state that is not possible to go below this value it's necessary to prove the converse theorem. The converse part shows that if we use an average codeword length even slightly below the entropy we are no longer able to decode.

Theorem (*Shannon's Source Coding: converse)***.**
Let $X \sim p(x)$ be a DMS with alphabet $\mathcal{X}$. Let us indicate by $P(\text{err})$ the probability of not being able to decode, that is the probability of incurring in a non invertible code. Then, for any $\nu > 0$, for any coding scheme $C(x^n)$ such that for large n $L/n = H(X) - \nu$, $\forall \delta > 0$

$$P(\text{err}) \geq 1 - \delta - 2^{-\frac{\nu n}{2}}. \tag{4.27}$$

Proof. Since the average number of bits used for a n-length sequence is $n(H(X) - \nu)$, we can encode at most $2^{n(H(X)-\nu)}$ sequences. Let us search for a good choice of the sequences to index; the best thing to do is trying to encode at least the sequences in $A_\varepsilon^{(n)}$. As to the non-typical sequences, if there are no bits left, we assign to each of them the same codeword (in this way the code loses the invertibility property but only for non-typical sequences). However, it is easy to argue that the number of sequences we can encode through this procedure is less than the total number of typical sequences. In order to show that, let us set $\varepsilon = \nu/2$, and then consider $A_{\nu/2}^{(n)}$. We evaluate the probability of a correctly encoded sequence[4](i.e. the probability of falling into the set of the correctly encoded sequences), namely $P(\text{corr})$, which has the following expression:

$$\begin{aligned} P(\text{corr}) &= \sum_{x^n \in A_{\nu/2}^{(n)}:\ x^n \leftrightarrow c(x^n)^5} p(x^n) \\ &\leq 2^{n(H(X)-\nu)} \cdot 2^{-n(H(X)+\nu/2)} = 2^{-n\frac{\nu}{2}}, \end{aligned} \tag{4.28}$$

where the number of elements of the sum was upper bounded by the cardinality of the typical set and $p(x^n)$ by the upper bound of the probability of a typical sequence. Then, by considering that $\forall \delta > 0$ and large n $Pr\{A_{\nu/2}^{(n)}\} \geq 1 - \delta$, the probability that the source emits a sequence $A_{\nu/2}^{(n)}$ which can not be correctly coded is

$$P(\text{err}) \geq 1 - \delta - 2^{-n\frac{\nu}{2}}. \tag{4.29}$$

Notice that relation (4.29) is actually an underestimation of the bound for $P(\text{err})$ since we have not considered the contribution of the non typical se-

[4] With the term 'correctly' we mean 'in an invertible way'.

[5] The double arrow indicates that the mapping is one to one.

quences to $P(\text{err})$ (which cannot be correctly decoded too). However, we know that they have an arbitrarily small probability for large n so they do not have a significant impact on $P(\text{err})$.

□

The couple of theorems just proved gives a rigorous formalization of our previous discussion. Accordingly, *the entropy $H(X)$ gives the measure of the number of bits per symbol required on the average to describe a source of information.*
Some considerations follow from Shannon's source coding theorem. First of all, the coding procedure used to prove the theorem is highly impractical. Secondly, Shannon asserts that is possible to reach the entropy as long as we take n sufficiently large, that is if we jointly encode long sequences of symbols. This is another issue that raises a number of practical problems.

4.3.2 Extension to the Sources with Memory

In Chapter 3.2 we discussed the sources with memory and defined the entropy rate. We now want to discuss the encoding limits, in terms of average length, for the sources with memory. We prove that for stationary sources the entropy rate takes the role of the entropy in the memoryless case, with some subtle differences.

Extended and Adjoint Sources

To discuss information-theoretic concepts it is often useful to consider *blocks* rather than individual symbols, each block consisting of k subsequent source symbols. If we let $\mathbb{X}_n$ denote the source of information (with memory) with alphabet $\mathcal{X}$, each such block can be seen as being produced by an *extended source* $\mathbb{X}_n^k$ with a source alphabet $\mathcal{X}^k$. Given the extended source, it is possible to define the correspondent memoryless source $\mathbb{X}_n^{k,*}$, named *adjoint source*, by confining the memory to k. Then, the k-length blocks drawn from the source $\mathbb{X}_n^k$ are independent of each other.

Before stating the coding theorem for sources with memory we give the following lemma.

Lemma (*Behavior of the average entropy).*
For any stationary source with memory $\mathbb{X}_n$ the sequence of values $\frac{H(X_k,...,X_1)}{k}$ tends to $\mathcal{H}(\mathbb{X}_n)$ from above as $k \to \infty$, that is for large k

$$\frac{H(X_k, ..., X_1)}{k} - \mathcal{H}(\mathbb{X}_n) \geq 0. \tag{4.30}$$

Proof. By applying the chain rule to the joint entropy twice, we have

$$\begin{aligned} H(X_k, ..., X_1) &= H(X_{k-1}, ..., X_1) + H(X_k|X_{k-1}, ..., X_1) \qquad (4.31)\\ &= H(X_{k-2}, ..., X_1) + H(X_{k-1}|X_{k-2}, ..., X_1) + \\ &\quad +H(X_k|X_{k-1}, ..., X_1). \qquad (4.32)\end{aligned}$$

The stationarity assumption permits to consider shifts of the random variables; hence, going on from (4.32)

$$\begin{aligned} &= H(X_{k-2}, ..., X_1) + H(X_k|X_{k-1}, .., X_2) + H(X_k|X_{k-1}, .., X_1) \qquad (4.33)\\ &\geq H(X_{k-2}, ..., X_1) + 2H(X_k|X_{k-1}, ..., X_1), \qquad (4.34)\end{aligned}$$

where inequality (4.34) is obtained by adding X_1 to the conditioning variables of the second entropy term in (4.33) (remember that conditioning reduces entropy). By iterating the same process $k-2$ more times we obtain

$$H(X_k, ..., X_1) \geq kH(X_k|X_{k-1}, ..., X_1). \qquad (4.35)$$

Deriving from relation (4.35) an upper bound for the conditional entropy $H(X_k|X_{k-1}, ..., X_1)$ and substituting it in the expression in (4.31) we get

$$\begin{aligned} H(X_k, ..., X_1) &= H(X_{k-1}, ..., X_1) + H(X_k|X_{k-1}, ..., X_1)\\ &\leq H(X_{k-1}, ..., X_1) + \frac{H(X_k, ..., X_1)}{k}. \qquad (4.36)\end{aligned}$$

Moving the term $H(X_k, ..., X_1)/k$ on the left-hand side of the inequality and dividing both sides by $k-1$ yields

$$\frac{H(X_k, ..., X_1)}{k} \leq \frac{H(X_{k-1}, ..., X_1)}{k-1}, \qquad (4.37)$$

which proves that the sequence of the mean entropies on k outputs is non-increasing with respect to k.
Then, from the definition of the entropy rate equation (4.30) follows. □

Theorem (*Source Coding with Memory: direct*).
Let $\mathbb{X}_n$ be a stationary source and let x^n be a sequence of n symbols emitted by the source. Then, $\forall \varepsilon > 0$ and for sufficiently large n, $\exists\ C(x^n)$ s.t.

$$\frac{L}{n} \leq \mathcal{H}(\mathbb{X}_n) + \varepsilon. \qquad (4.38)$$

Proof. Let us consider the k-th order extension of $\mathbb{X}_n$, i.e. the source $\mathbb{X}_n^k$ having alphabet $\mathcal{X}^k$. Let $\mathbb{X}_n^{k,*}$ denote the adjoint source, i.e. the corresponding discrete memoryless source that we get by confining the memory to a length k[6]. We can apply the noiseless source coding theorem to $\mathbb{X}_n^{k,*}$. According to the direct theorem we have that: $\forall \varepsilon > 0$, $\exists N_0 : \forall N > N_0 \; \exists C(x^{(k,*),N})$ s.t.

$$\frac{E[l(x^{(k,*),N})]}{N} \leq H(X_k, X_{k-1}, ..., X_1) + \varepsilon, \tag{4.39}$$

where $x^{(k,*),N}$ denotes a N-length sequence of blocks drawn from the memoryless source. According to the entropy rate definition, for $k \to \infty$

$$H(X_k, X_{k-1}, ..., X_1) \to \mathcal{H}(\mathbb{X}_n) \cdot k. \tag{4.40}$$

By the definition of the entropy rate, we know that for any positive number δ, if k is large enough,

$$\frac{H(X_k, X_{k-1}, ..., X_1)}{k} \leq \mathcal{H}(\mathbb{X}_n) + \delta. \tag{4.41}$$

Substituting (4.41) in (4.39) yields

$$\frac{E[l(x^{(k,*),N})]}{N} \leq k \cdot \mathcal{H}(\mathbb{X}_n) + k \cdot \delta + \varepsilon. \tag{4.42}$$

Since $\frac{E[l(x^{(k,*),N})]}{N}$ is the *average number of bits per block*, we can divide by k in order to obtain the average number of bits per symbol. Then,

$$\frac{E\left[l(x^{(k,*),N})\right]}{k \cdot N} \leq \mathcal{H}(\mathbb{X}_n) + \delta + \frac{\varepsilon}{k}. \tag{4.43}$$

The product $k \cdot N$ is the total length of the starting sequence of symbols, i.e. $k \cdot N = n$. Thus, setting $\varepsilon' = \delta + \frac{\varepsilon}{k}$ we have

$$\frac{E[l(x^n)]}{n} \leq \mathcal{H}(\mathbb{X}_n) + \varepsilon, \tag{4.44}$$

and the theorem is proved.

□

An important aspect which already comes out from the direct theorem is that, in order to reach the entropy rate, we have to encode blocks of sym-

[6] It must be pointed out that the joint entropy $H(X_k, X_{k-1}, ..., X_1)$ is not the sum of the single entropies because of the presence of memory among the symbols within the block.

bols. However, with respect to the memoryless source coding we have now two parameters, the block length k and the number of blocks N, which have both to be large (tend to infinity) to approach the entropy rate.
We now consider the converse theorem.

Theorem (*Source Coding with Memory: converse)***.**
Given a stationary source $\mathbb{X}_n$, the average number of bits per symbol required by an invertible code can not be less than the entropy rate $\mathcal{H}(\mathbb{X}_n)$.

Proof. The proof is given by contradiction. Let us suppose that for large enough n it is possible to encode with a rate less then $\mathcal{H}(\mathbb{X}_n)$, say $\mathcal{H}(\mathbb{X}_n) - \nu$ for any arbitrarily small $\nu > 0$. Given such a code, we can apply the same mapping to the memoryless source $\mathbb{X}_n^{k,*}$. In other words, given a sequence drawn from the source $\mathbb{X}_n^{k,*}$, we consider it as if it were generated from $\mathcal{H}(\mathbb{X}_n)$ and we assign it the correspondent codeword. Then, we get the following expression for the average number of bits per block,

$$\begin{aligned} E\left[l(x^{(k,*)})\right] &= k \cdot (\mathcal{H}(\mathbb{X}_n) - \nu) \\ &\leq k \cdot \frac{H(X_k, ..., X_1)}{k} - k \cdot \nu, \\ &= H(X_k, ..., X_1) - k \cdot \nu, \end{aligned} \tag{4.45}$$

where the inequality follows from the Lemma stating the behavior of the average entropy. By looking at equation (4.45) we realize the expected contradiction. Equation (4.45) says that it is possible to code the output of a DMS, namely $\mathbb{X}_n^{k,*}$, at a rate lower then the entropy. This fact is in contrast with the noiseless source coding theorem.

□

4.4 Data Compression

Ever since Shannon proved the noiseless coding theorem, researchers tried develop practical codes which achieve Shannon's limit.
We know from the previous section that any practical source code must be uniquely decodable. Below, we define a more stringent property

Property (Instantaneous code)**.** A uniquely decodable code is said to be a *prefix code* or *instantaneous code* if no codeword is prefix[7].

[7] A codeword c_1 is prefix of another codeword c_2 when the string of bits of the first codeword matches exactly the first $l(c_1)$ bits of the second codeword.

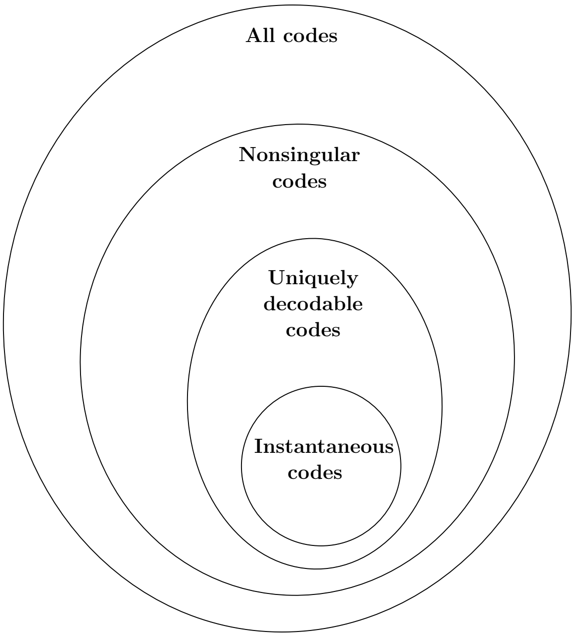


Figure 4.2: Relation among the classes of codes.

X	Singular	Nonsingular, Not Uniquely Decodable	Uniquely Decodable, Not instantaneous	Instantaneous
a	0	0	1	0
b	1	00	10	10
c	11	1	100	110
d	0	11	1000	111

Table 4.1: Example of cedes.

If a code is instantaneous it is possible to decode each codeword in a sequence without reference to succeeding code symbols. This allows the receiver to decode immediately at the end of the codeword, thus reducing the decoding delay. In practice, we always use instantaneous codes. Figure 4.2 illustrates the relations between the classes of codes. Some examples of various kinds of codes are given in Table 4.1.
In order to highlight the differences between the analysis developed in this chapter and Shannon's source coding theorem, we remind that Shannon refers, more in general, to uniquely decodable codes (indeed, he adopt non singular block codes).

4.4.1 Kraft Inequality

In designing codes, our concern is the length of the codewords rather than the specific codewords, being the length the parameter which affects the transmission rate and the storage requirements. Our goal is indeed *to design instantaneous codes with the minimum length.* From the definition of prefix codes it is easy to argue that it may not always be possible to find a prefix code given a set of codeword lengths. For instance, it is easy to guess that it is not possible to design a prefix code with lengths 2,3,2,2,2 (we are forced to use all the couples 00,01,10,11 already for the four two-length codewords...).
The question is then what codeword lengths can we use to design prefix codes?

Theorem (*Kraft's inequality*)**.**
A necessary and sufficient condition for the existence of a prefix code with codeword lengths $l_1, l_2, ..., l_{|\mathcal{X}|}$ is that

$$\sum_{i=1}^{|\mathcal{X}|} 2^{-l_i} \leq 1. \tag{4.46}$$

Proof. Consider a binary tree, as the one depicted in Figure 4.3, in which each node has 2 children. The branches of the tree represents the symbols of the codeword, 0 or 1. Then, each codeword is represented by a node or a leaf on the tree. The path from the root traces out the symbols of the codeword. The property of prefix code implies that in the tree no codeword is an ancestor of any other codeword, that is, the presence of a codeword eliminates all its descendants as possible codewords. Then, for a prefix code, each codeword is represented by a leaf.

- (*Necessary condition*)*: for any instantaneous code, the codeword lengths satisfy the Kraft inequality.*

Let l_{max} be the length of the longest codeword (i.e. the depth of the tree). A codeword at level l_i has $2^{l_{max}-l_i}$ descendants at level l_{max}, which cannot be codewords of a prefix code and must then be removed from the tree. For a prefix codes with the given lengths $(l_1, l_2, ..., l_{|\mathcal{X}|})$ to exist, the overall number of leaves that we remove from the tree must be less than those available

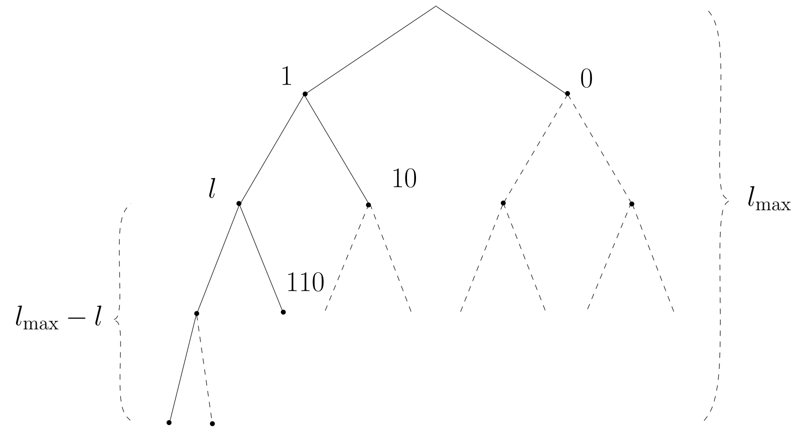


Figure 4.3: Code tree for Kraft inequality.

($2^{l_{max}}$). In formula:

$$\sum_{i=1}^{|\mathcal{X}|} 2^{l_{max}-l_i} \leq 2^{l_{max}}, \tag{4.47}$$

which, divided by $2^{l_{max}}$, yields

$$\sum_{i=1}^{|\mathcal{X}|} 2^{-l_i} \leq 1. \tag{4.48}$$

- (*Sufficient condition*): *given a set of lengths satisfying Kraft's inequality there exists an instantaneous code with these codeword lengths.*

Let us construct the code with the given set of lengths. For each length l_i, we consider $2^{l_{max}-l_i}$ leaves of the tree, label the root of the correspondent subtree (which corresponds to a node at depth l_i) as the codeword i and remove all its descendants from the tree. This procedure can be repeated for all the lengths if there are enough leaves, that is if

$$\sum_{i} 2^{l_{max}-l_i} \leq 2^{l_{max}}. \tag{4.49}$$

□

If Kraft's inequality holds, then we can construct a code which is prefix. Then, Kraft's inequality tells us if, given a set of codeword lengths, a prefix code exists. Note that, however, it does not tell us whether a code satisfying it is instantaneous.

Given a source $X \sim p(x)$, we seek the *minimum average length* of a prefix code for the source ($C : \mathcal{X} \to \{0,1\}^*$). To do so, we have to solve the following constrained optimization problem:

$$\min_{l(x)} \sum_{x \in \mathcal{X}} p(x) l(x), \tag{4.50}$$

subject to

$$\begin{aligned} \sum_x 2^{-l(x)} &\leq 1, \\ l(x) &\in \mathbb{N}. \end{aligned} \tag{4.51}$$

Minimization (4.50) subject to the constraints in (4.51) is hard to solve. In the next section we see, for a particular choice of codeword lengths, how far the average length remains from the minimum value (i.e. $H(X)$).

4.4.2 Alternative proof of Shannon's source coding theorem for instantaneous codes

It is interesting to compare the minimum average length L for instantaneous codes with the entropy of the source. We know from Shannon's theorem that L is surely greater or at most equal to the entropy.

Property. For a diadic source[8] X it is possible to build an instantaneous code with average length $L = H(X)$.

[8] In a diadic source the probability of the symbols are negative quadratic powers of 2, that is $p_i = 2^{-\alpha_i}$, ($\alpha_i \in \mathbb{N}$).

Proof.

$$\begin{aligned} L - H(X) &= \sum_i p_i l_i - \sum_i p_i \log p_i \\ &= \sum_i p_i \log 2^{l_i} + \sum_i p_i \log p_i \\ &= \sum_i p_i \log \frac{p_i}{2^{-l_i}} \\ &\geq \log e \sum_i p_i \left(1 - \frac{2^{-l_i}}{p_i}\right) \\ &= \log e(\underbrace{\sum_i p_i}_{=1} - \sum_i 2^{-l_i}) \geq 0. \end{aligned} \tag{4.52}$$

If the source is diadic, $l_i = \log_2(1/p_i)$ belongs to $\mathbb{N}$ for each i, and then, by using these lengths for the codewords, the derivation in (4.52) holds at equality. □

What if the source is not diadic? The following property tells us how far from $H(X)$ the minimum average codeword length is (at most) in the general case.

Theorem (*Average length)*.
For any source X, there exists a prefix code with average length satisfying

$$H(X) \leq L \leq H(X) + 1. \tag{4.53}$$

Proof. The left-hand side has already been proved in (4.52). In order to prove the right-hand side, let us assign the lengths l_i according to a round-off approach, i.e. by using the following approximation:

$$l_i = \lceil \log \frac{1}{p_i} \rceil \leq \log \frac{1}{p_i} + 1. \tag{4.54}$$

The average codeword length of this code is

$$L = \sum_i p_i l_i \leq \sum_i p_i \left(\log \frac{1}{p_i} + 1\right) = H(X) + 1. \tag{4.55}$$

Since this code, built by means of the round-off approximation, is only a

particular choice, for the minimum-length code we surely have $L \leq H(X)+1$.
□

So far we have assumed to encode each source symbol separately. According to Shannon's theorem, to get closer to the entropy we must encode together blocks of symbols. Let X^k denote the extended source. Now, C maps each k-length block of source symbols into a string of bits, that is $C : \mathcal{X}^k \to (0,1)^*$. Then, the average codeword length in bits per symbol is L_k/k, being L_k the average codeword length for the k-th extended source.

Theorem (*Instantaneous Source Coding)*.
For a memoryless source X, there exists an instantaneous code with average length L_k satisfying

$$H(X) \leq \frac{L_k}{k} \leq H(X) + \frac{1}{k}. \tag{4.56}$$

Proof. Let L_k^* be the minimum average length for a code of the extended source X_k. Applying the theorem on the average length to the extended source yields

$$H(X^k) \leq L_k^* \leq H(X^k) + 1. \tag{4.57}$$

Being the source memoryless $H(X^k) = kH(X)$, if we consider the average number of bits per symbol spent for the encoding, i.e. L_k^*/k, equation (4.56) is proved.

□

As expected, for any source X when $k \to \infty$ we have that $L \to H(X)$. Then, requiring the code to be instantaneous (more than uniquely decodable) does not change the minimum number of bits per symbol we have to spend for the lossless encoding of the source. Again, in order to reach the entropy value $H(X)$ for a generic source we have to code long blocks of symbols and not separate symbols. The theorem can then be seen as a formulation of the Shannon coding theorem for instantaneous codes.

Coding of Sources with Memory

We want to restate the above theorem for the sources with memory.
Let $\mathbb{X}_n$ be a stationary source and $\mathbb{X}_n^k$ the correspondent extended source. Let L_k denote the average length of a code for the extended source.

Theorem (*Instantaneous Coding for Sources with Memory)*.
Given a source with memory $\mathbb{X}_n$, there exists an instantaneous code for its kth extension satisfying

$$\mathcal{H}(\mathbb{X}_n) \leq \frac{L_k}{k} \leq \mathcal{H}(\mathbb{X}_n) + \varepsilon + \frac{1}{k}. \tag{4.58}$$

where ε is a positive quantity which can be taken arbitrarily small for large k.

Proof. If we consider the adjoint source $\mathbb{X}_n^{k,*}$, we have the following bounds for the length of the optimum code:

$$H(X^{k,*}) \leq L_k \leq H(X^{k,*}) + 1, \tag{4.59}$$

and then

$$H(X_k, ..., X_1) \leq L_k \leq H(X_k, X_{k-1}, ..., X_1) + 1. \tag{4.60}$$

Dividing by k yields

$$\frac{H(X_k, ..., X_1)}{k} \leq \frac{L_k}{k} \leq \frac{H(X_k, ..., X_1)}{k} + \frac{1}{k}. \tag{4.61}$$

The proof of the lower bound in (4.58) directly follows from the Lemma on the behavior of the average entropy in Section 4.3.2 ($\frac{H(X_k,...,X_1)}{k} \geq \mathcal{H}(\mathbb{X}_n)$), while in order to prove the upper bound we exploit the definition of the entropy rate: for any $\varepsilon > 0$, there exists k such that $\frac{H(X_k,...,X_1)}{k} \leq \mathcal{H}(\mathbb{X}_n) + \varepsilon$. Then, relation (4.58) holds.

□

From the above theorem it is evident that the benefit of coding blocks of symbols is twofold:

- the round off to the next integer number, which costs 1 bit, is spread on k symbols (this is the same benefit we had for memoryless sources);
- the ratio $\frac{H(X_k,...,X_1)}{k}$ decreases as k increase (while in the memoryless case $\frac{H(X_k,...,X_1)}{k} = H(X)$).

Therefore, we argue that coding blocks of symbols rather than individual symbols is even more necessary when we deal with sources with memory, since it leads to a great gain in terms of bits saving.

Chapter 5

Channel Capacity and Coding

In the previous chapters we faced with the problem of source coding. Once encoded, the information must be transmitted through a *communication channel* to reach its destination. This chapter is devoted to the study of this second step of the communication process.

5.1 Discrete Memoryless Channel

Each communication channel is characterized by the relation between the input and the output. For simplicity, throughout the analysis, we consider only discrete time channels. We know that, from an information theory perspective, the signals carry information and then they have a random nature; specifically they are stochastic processes $x(k,t)$. According to Shannon's sampling theorem, which also holds for random signals, if the signal bandwidth is limited, we can consider its samples[1] and then we can assume that the channel is discrete in time. The sampling of the stochastic process yields at the input of the channel the sequence of random variables $x(k,nT)$, as depicted in Figure 5.1. To ease the notation, we refer to the sequence $x(k,nT)$ as a sequence of random variables X_n, omitting the dependence on k. Clearly, the channel input can be seen as the outcome of an information source.
As to the values assumed by each random variable X_n, if the input source has a finite alphabet ($|\mathcal{X}| < \infty$) we have a discrete channel, a continuous channel otherwise.

[1] As a matter of fact the requirement of limited bandwidth is not necessary due to the presence of the channel which acts itself as bandwidth limiter.

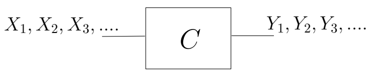


Figure 5.1: Discrete time channel. The input sequence is the sampling the stochastic process $x(k, t)$ with sampling step T.

5.1.1 A Mathematical Model for the channel

There are many factors, several of which with a random nature, that in a physical channel cause the output to be different from the input, e.g. attenuation, multipath, noise. Then, the input-output relation in a communication channel is, generally, a stochastic relation.

Definition. A *discrete channel* is a statistical model with an input X_n and an output Y_n which can be seen as a *noisy* version of X_n. The sequences X_n and Y_n take value in $\mathcal{X}$ and $\mathcal{Y}$ respectively ($|\mathcal{X}|, |\mathcal{Y}| < \infty$).

Given the input and the output alphabet $\mathcal{X}$ and $\mathcal{Y}$, a *channel* is described by the probabilistic relationship between the input and the output, i.e. by the set of *transition probabilities*

$$Pr\{Y_k = y | X_1 = x_1, X_2 = x_2, ..., X_k = x_k\} \quad y \in \mathcal{Y}, (x_1, ..., x_k) \in \mathcal{X}^k \quad (5.1)$$

where k denotes the discrete time at which the outcome is observed. Note that, due to causality, conditioning is restricted to the inputs preceding k and to the k-th input itself.

The channel is said *memoryless* when the output symbol at a given time depends only on the current input. In this case the transition probabilities become:

$$Pr\{Y_k = y | X_k = x\} \quad \forall y \in \mathcal{Y}, \forall x \in \mathcal{X}. \quad (5.2)$$

and the simplified channel scheme is illustrated in Figure 5.2. Assuming a memoryless channel greatly restricts our model since in this way we do not consider several factors, like fading, which could affect the communication because due to the introduction of intersymbol interference. Such phenomena require the adoption of much more complex models.

In order to further simplify the analysis, we also assume that the channel is *stationary*. Frequently[2], we can make this assumption without loss of generality since the channel variability is slow with respect to the transmission rate. In other words, during the transmission of a symbol, the statistical properties of the channel do not change significantly. Then, since the probabilistic

[2] This is not true when dealing with mobile channels.

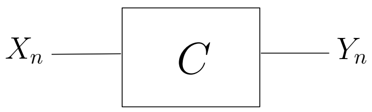


Figure 5.2: Discrete memoryless channel. The output signal at each time instant n (r.v.) depends on the input signal (r.v.) at the same time.

model describing the channel does not change over time, we can characterize the channel by means of the transition probabilities $p(y|x)$, where $y \in \mathcal{Y}$ and $x \in \mathcal{X}$. These probabilities can be conveniently arranged in a matrix $\mathbf{P} = \{P_{ij}\}$, where

$$P_{ij} = P\{y_j|x_i\} \quad j = 1, .., |\mathcal{Y}| \quad i = 1, ..., |\mathcal{X}|. \tag{5.3}$$

The matrix $\mathbf{P}$ is called *channel matrix* or *channel transition matrix.*

5.1.2 Examples of discrete memoryless channels

Noiseless binary channel

Suppose that we have a channel in which the binary input is reproduced exactly at the output. Then, any transmitted bit is received without error. The transition matrix is

$$\mathbf{P} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{5.4}$$

This is a limit case, for which we have no longer a probabilistic channel. A graphical representation of the noiseless channel is given in Figure 5.3.

Noisy channel with non-overlapping outputs

This is another example in which noise does not affect the transmission, even if the channel is probabilistic. Indeed, see Figure 5.4, the output of the channel depends randomly on the input; however the input can be exactly determined from the output and then every transmitted bit can be recovered without any error. The transition matrix is

$$\mathbf{P} = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \end{pmatrix}. \tag{5.5}$$

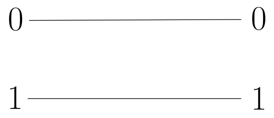


Figure 5.3: Noiseless binary channel.

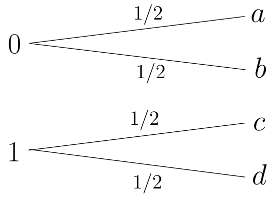


Figure 5.4: Model of the noisy channel with non overlapping outputs.

Noisy Typewriter

This is a more realistic example. A channel input is delivered unchanged at the output with probability 1/2 and transformed into the subsequent element with probability 1/2. In this case, the transmitted signal can not be correctly recovered from the output. Figure 5.5 illustrates the behavior of this channel; the transition matrix has the following form

$$\mathbf{P} = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & ... & 0 \\ 0 & 1/2 & 1/2 & 0 & ... & 0 \\ ... & ... & ... & ... & ... & 0 \\ 1/2 & 0 & ... & ... & ... & 1/2 \end{pmatrix}. \tag{5.6}$$

Binary Symmetric Channel (BSC)

The binary symmetric channel is a binary channel in which the input symbols are flipped with probability ε and left unchanged with probability $1 - \varepsilon$ (Figure 5.6). The transition matrix of the BSC is

$$\mathbf{P} = \begin{pmatrix} 1-\varepsilon & \varepsilon \\ \varepsilon & 1-\varepsilon \end{pmatrix}. \tag{5.7}$$

This channel model is used very frequently in communication engineering. Without loss of generality, we will only consider BSC with $\varepsilon < 1/2$. Indeed, if $\varepsilon > 1/2$, we can trivially reverse the input symbols thus yielding an error probability lower than 1/2.

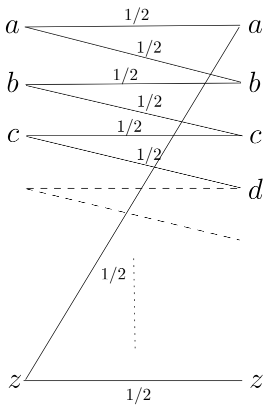


Figure 5.5: Noisy typewriter.

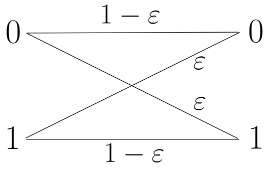


Figure 5.6: Binary symmetric channel.

Binary Erasure Channel (BEC)

This channel is similar to the binary symmetric channel, but in this case the bits are lost, rather than flipped, with a given probability α. The transition matrix is

$$\mathbf{P} = \begin{pmatrix} 1-\alpha & \alpha & 0 \\ 0 & \alpha & 1-\alpha \end{pmatrix}. \tag{5.8}$$

The channel model is depicted in Figure 5.7.

5.2 Channel Coding

From previous chapters we know that $H(X)$ represents the fundamental limit on the rate at which a discrete memoryless source can be encoded. We we will prove that a similar fundamental limit also exists for the transmission rate over communication channels.

The main goal when transmitting information over any communication channel is *reliability*, which is measured by the probability of correct reception at the output of the channel. The surprising result that we will prove in this

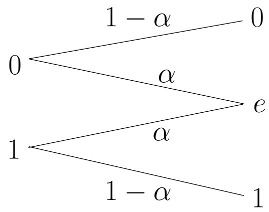


Figure 5.7: Binary erasure channel.

chapter is that reliable transmission is possible even over noisy channels, as long as the transmission rate is sufficiently low. The existence of a fundamental bound on the transmission rate, proved by Shannon, is one of the most remarkable results of information theory.
By referring to the example of the noisy typewriter in Section 5.1.2, some interesting considerations can be made. By using only half of the inputs, it is possible to make the corresponding outputs disjoint, and then recover the input symbols from the output. Then, this subset of the inputs can be transmitted over the channel with no error. This is just an example in which the limitation that the noise causes in the communication is not on the reliability of the communication but on the rate of the communication. This example provides also a first insight into channel coding: limiting the inputs to a subset is similar to the addition of redundancy which will be performed through channel coding.

5.2.1 Preview of the Channel Coding Theorem

BSC: a qualitative analysis

By looking at the binary symmetric channel we try to apply a similar approach to that used for the noisy typewriter in order to determine if non-overlapping outputs, and then transmission without error, can be obtained in the BSC case. To this purpose, we have to consider sequences of input symbols instead of single inputs. Then, we define the n-th extension of the channel or *extended channel*, which is a channel having input and output alphabets $\mathcal{X}^n = \{0,1\}^n$ and $\mathcal{Y}^n = \{0,1\}^n$ and transition probabilities $p(y^n|x^n) = \prod_{i=1}^{n} p(y_i|x_i)$. Figure 5.8 gives a schematic representation of the extended channel. Due to the dispersion introduced by the channel, a set of possible output sequences corresponds to a n-length transmitted sequences. If the sets corresponding to different input sequences were disjoint, the transmission would be error-free. This happens only with channels having non-overlapping outputs. By looking at the BSC, Figure 5.6, we see that it is no so, but we can consider a subset of the input sequences in order

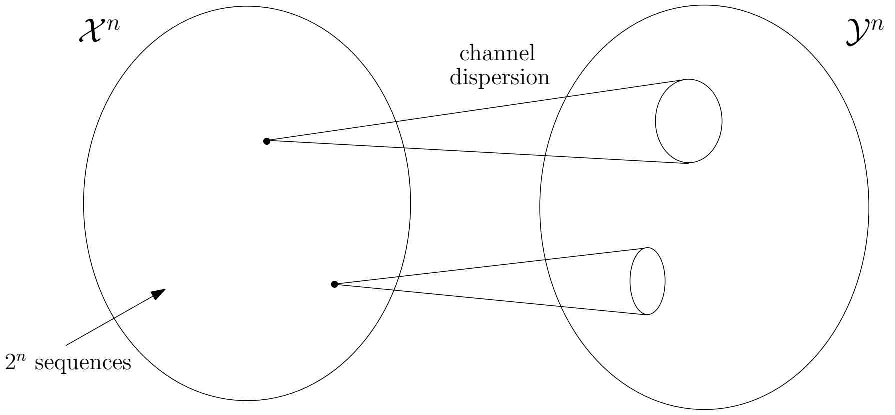


Figure 5.8: Representation of the n-th extension of the channel.

to make the corresponding set disjoint. That is, we can consider 2^k input sequences for some value k ($k < n$). Note that, without noise, k bits would suffice to index 2^k sequences; the $n - k$ additional bits in each sequence correspond to the 'redundancy'. In the sequel we better formalize this concept.

In the BSC, according to the law of large numbers, if a binary sequence of length n (for large n) is transmitted over the channel with high probability, the output will disagree with the input at about $n\varepsilon$ positions. The number of possible ways in which it is possible to have $n\varepsilon$ error in a n-length sequence (or the number of possible sequences that disagree with the input in $n\varepsilon$ positions) is given by

$$\binom{n}{n\varepsilon}. \tag{5.9}$$

By using Stirling's approximation $n! \approx n^n e^{-n}\sqrt{2\pi n}$ and by applying some algebra we obtain

$$\binom{n}{n\varepsilon} \approx \frac{2^{nh(\varepsilon)}}{\sqrt{2\pi n(1-\varepsilon)\varepsilon}}. \tag{5.10}$$

Relation (5.10) gives an approximation on the number of sequences in each output set. Then, for each block of n inputs, there exist roughly $2^{nh(\varepsilon)}$ highly probable corresponding output blocks. Note that if $\varepsilon = 1/2$, then $h(\varepsilon) = 1$ and the entire output set would be required for an error-free transmission of only one input sequence.

On the other hand, by referring to the output of the channel, regarded as a source, the total number of highly probable sequences is roughly $2^{nH(Y)}$. Therefore, the maximum number of input sequences that may produce almost non-overlapping output sets is at most equal to

$$M = \frac{2^{nH(Y)}}{2^{nh(\varepsilon)}/\sqrt{2\pi n(1-\varepsilon)\varepsilon}}. \tag{5.11}$$

As a consequence, the maximum number of *information bits* that can be correctly transmitted is

$$k = \log_2\left(2^{n(H(Y)-h(\varepsilon))} \cdot \sqrt{2\pi n(1-\varepsilon)\varepsilon}\right). \tag{5.12}$$

Then, the number of bit that can be transmitted each time, i.e. the transmission rate for channel use is:

$$R = \frac{k}{n} = \frac{\log_2(2^{n(H(Y)-h(\varepsilon))} \cdot \sqrt{2\pi n(1-\varepsilon)\varepsilon})}{n}. \tag{5.13}$$

Finally, as $n \to \infty$, $R \to H(Y) - h(\varepsilon)$.
A close inspection of the limit expression for R reveals that we have still a degree of freedom that can be exploited to maximize the transmission rate; it consists in the input probabilities $p(x)$, which determine the values of $p(y)$ (remember that the transition probability of the channel are fixed by the stationarity assumption) and then $H(Y)$. In the sequel we look for the input probability distribution maximizing $H(Y)$, giving the maximum transmission rate. Since Y is a binary source, the maximum of $H(Y)$ is 1, which is obtained when the input symbols are equally likely. So, the maximum transmission rate is $R_{max} = 1 - h(\varepsilon)$.

Observation.
The quantity $1-h(\varepsilon)$ is exactly the maximum value of the mutual information between the input and the output for the binary symmetric channel (BSC), that is

$$\max_{p_X(x)} I(X;Y) = 1 - h(\varepsilon). \tag{5.14}$$

In fact, given the input bit x, the BSC behaves as a binary source, giving at the output the same bit with probability $1-\varepsilon$. Thus, we can state that $H(Y|X) = h(\varepsilon)$ and consequently $I(X;Y) = H(Y) - H(Y|X) = H(Y) - h(\varepsilon)$, whose maximum is indeed $1 - h(\varepsilon)$.

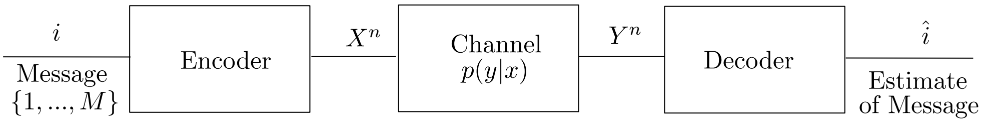


Figure 5.9: Communication channel.

Qualitative analysis of a general discrete memoryless channel

The previous analysis explains the essence of Shannon's theorem on the channel coding by focusing specifically on the binary symmetric channel. In order to extend the previous analysis to a generic channel we need some clarifications. Firstly, we note that when we refer to sets of outputs we do not mean necessarily a compact set. Given an input, the corresponding output sequence may be scattered throughout the whole space $\mathcal{Y}^n$, depending on the behavior of the channel. Secondly, the output sets in a general channel have usually different sizes since the channel is not symmetric.

We can affirm that, given an input sequence x^n, the number of possible output sequences y^n is approximately $2^{nH(Y|x^n)}$, with high probability. This is indeed the approximate number of typical sequences with respect to the distribution $p(y|X^n = x^n)$. By varying the input sequence x^n, we can consider the mean number of output sequences $2^{nH(Y|X)}$. Since the total number of typical sequences for the source Y is still $2^{nH(Y)}$, it follows that the maximum number of disjoint sets is $2^{n(H(Y)-H(Y|X))} = 2^{nI(X;Y)}$. Accordingly, we can correctly transmit $I(X;Y)$ information bits for channel use. By properly choosing the prior probabilities, we directly have the following expression for the maximum achievable rate:

$$R_{max} = \max_{p_X(x)} I(X;Y). \tag{5.15}$$

This result is in agreement with the previous one for the BSC. We foretell that the above expression represents the channel capacity.

In Section 5.2.3, we will give a rigorous formalization to the above considerations by proving the *noisy channel-coding theorem.*

5.2.2 Definitions and concepts

Let $\{1, 2, ..., M\}$ be the index set from which a message is drawn. Before being transmitted into the channel the indexes are encoded. At the receiver side, by observing the output of the channel the receiver guesses the index through an appropriate decoding rule. The situation is depicted in Figure 5.9. Let us rigorously define some useful concepts, many of them already discussed in the previous section.

Definition. A *discrete memoryless channel* (DMC) consists of two finite sets $\mathcal{X}$ and $\mathcal{Y}$ and a collection of probability mass functions $p(y|x)$, denoted by $(\mathcal{X}, p(y|x), \mathcal{Y})$.

Definition. The *nth extension of the discrete memoryless channel* corresponds to the channel $(\mathcal{X}^n, p(y^n|x^n), \mathcal{Y}^n)$, where

$$p(y_k|x^k, y^{k-1}) = p(y_k|x_k), \quad k = 1, 2, ..., n \tag{5.16}$$

i.e. the output does not depend on the past inputs and outputs.
If the channel is used *without feedback*, i.e. if the input symbols do not depend on the past output symbols ($p(x_k|x^{k-1}, y^{k-1}) = p(x_k|x^{k-1})$), the channel transition probabilities for the nth extension of the DMC can be written as

$$p(y^n|x^n) = \prod_{i=1}^{n} p(y_i|x_i). \tag{5.17}$$

We shall always implicitly refer to channels without feedback, unless stated otherwise.

Definition. An (M, n) code for the channel $(\mathcal{X}, p(y|x), \mathcal{Y})$ consists of:
1. An encoding function $g : \{1 : M\} \to \mathcal{X}^n$, which is a mapping from the index set to a set of codewords or *codebook*.
2. A decoding function $f : \mathcal{Y}^n \to \{1 : M\}$, which is a deterministic rule assigning a number (index) to each received vector.

Definition. Let λ_i be the error probability given that index i was sent, namely the *conditional probability of error*:

$$\lambda_i = Pr\{f(y^n) \neq i|x^n = g(i)\}. \tag{5.18}$$

Often, we will use $x^n(i)$ instead of $g(i)$ to indicate the codeword associated to index i. As a consequence of the above definition, the *maximal probability of error* $\lambda_{max}^{(n)}$ for an (M, n) code is defined as

$$\lambda_{max}^{(n)} = \max_{i \in \{1,2,...,M\}} \lambda_i. \tag{5.19}$$

The *average probability of error* $P_e^{(n)}$ for an $(M.n)$ code is

$$P_e^{(n)} = \frac{1}{M} \sum_{i=1}^{M} \lambda_i, \tag{5.20}$$

where we implicitly assumed that the indexes are drawn in an equiprobable manner. We point out that the average probability of error, like the maximum one, refers to the n-length sequences.

Definition. The *rate* R of an (M, n) code is

$$R = \frac{\log M}{n} \quad \text{bits per channel use.} \tag{5.21}$$

Definition. A rate R is said to be *achievable* if there exists a sequence of codes having rate R, i.e. $(2^{nR}$, $n)$ codes, such that

$$\lim_{n \to \infty} \lambda_{max}^{(n)} = 0. \tag{5.22}$$

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

Jointly typical sequences and set

In order to describe the decoding process in Shannon's coding theorem it is necessary to introduce the concept of 'joint typicality'.

Definition. Given two DMSs X and Y, the set $A_\varepsilon^{(n)}$ of *joint typical* sequences $\{(x^n, y^n)\}$ with respect to the distribution $p(x, y)$, is the following set of n-long sequences

$$\begin{aligned} A_\varepsilon^{(n)} = &\{(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \\ &\left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \varepsilon, \left| -\frac{1}{n} \log p(y^n) - H(Y) \right| < \varepsilon, \\ &\left| -\frac{1}{n} \log p(x^n, y^n) - H(X, Y) \right| < \varepsilon \Bigg\}, \end{aligned} \tag{5.23}$$

where the first and the second conditions require the typicality of the sequences x^n and y^n respectively, and the last inequality requires the joint typicality of the couple of sequences (x^n, y^n).

We observe that if we do not considered the joint typicality, the number of possible couples of sequences in $A_\varepsilon^{(n)}$ would be the product $|A_{\varepsilon,x}^{(n)}| \cdot |A_{\varepsilon,y}^{(n)}| \cong 2^{n[H(X)+H(Y)]}$. The intuition suggests that the total number of jointly typical sequences is approximately $2^{nH(X,Y)}$ and then not all pairs of typical x^n and typical y^n are jointly typical since $H(X, Y) \leq H(X) + H(Y)$. These considerations are formalized in the following theorem, which is the extension of the AEP theorem to the case of two sources.

Theorem *(joint AEP)***.**
Let X and Y be two DMS with marginal probabilities p_X and p_Y and let (x^n, y^n) be a couple of sequences of length n drawn from the two sources. Then:

1. $\Pr\{A_\varepsilon^{(n)}\} \to 1 \quad \text{as} \quad n \to \infty \quad (> 1 - \delta \quad \text{for large } n)$;

2. $\forall \varepsilon, \quad |A_\varepsilon^{(n)}| \leq 2^{n(H(X,Y)+\varepsilon)} \quad \forall n$;

3. $\forall \delta, \forall \varepsilon, n$ large, $\quad |A_\varepsilon^{(n)}| \geq (1 - \delta) 2^{n(H(X,Y)-\varepsilon)}$;

4. Considering two sources $\tilde{X}$ and $\tilde{Y}$ with alphabets $\mathcal{X}$ and $\mathcal{Y}$ such that $p_{\tilde{X}} = p_X$ and $p_{\tilde{Y}} = p_Y$ but independent of each other, i.e. such that $(\tilde{X}^n, \tilde{Y}^n) \sim p_X(x^n) p_Y(y^n)$, we have
$$Pr\{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}\} \cong 2^{-nI(X;Y)}. \tag{5.24}$$
Formally,
$$\forall \varepsilon < 0, \forall n, \quad Pr\{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}\} \leq 2^{-n(I(X;Y)-3\varepsilon)} \quad , \tag{5.25}$$
and
$$\forall \varepsilon > 0, \forall \delta > 0, n \text{ large}, \quad Pr\{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}\} \geq (1 - \delta) 2^{-n(I(X;Y)+3\varepsilon)}. \tag{5.26}$$

Proof. The first point says that for large enough n, with high probability, the couple of sequences (x^n, y^n) lies in the typical set. It directly follows from the weak law of large numbers. In order to prove the second and the third point we can use the same arguments of the proof of the AEP theorem. Instead, we explicitly give the proof of point 4 which represents the novelty with respect to the AEP theorem. The new sources $\tilde{X}^n$ and $\tilde{Y}^n$ are independent but have the same marginals as X^n and Y^n, then

$$\begin{aligned} Pr\{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}\} &= \sum_{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}} p_{\tilde{X}}(\tilde{x}^n) p_{\tilde{Y}}(\tilde{y}^n) \\ &= \sum_{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}} p_X(\tilde{x}^n) p_Y(\tilde{y}^n) \\ &= \sum_{(x^n, y^n) \in A_\varepsilon^{(n)}} p_X(x^n) p_Y(y^n) \\ &\overset{(a)}{\leq} |A_\varepsilon^{(n)}| \cdot 2^{-n(H(X)-\varepsilon)} 2^{-n(H(Y)-\varepsilon)} \\ &\overset{(b)}{\leq} 2^{-n(H(X,Y)+\varepsilon)} 2^{-n(H(X)-\varepsilon)} 2^{-n(H(Y)-\varepsilon)} \\ &= 2^{-n(I(X;Y)-3\varepsilon)}, \end{aligned} \tag{5.27}$$

where inequality (a) follows from the AEP theorem, while (b) derives from point 2. Similarly, it's possible to find a lower bound for sufficiently large n, i.e.

$$\begin{aligned} Pr\{(\tilde{x}^n, \tilde{y}^n) \in A_\varepsilon^{(n)}\} &= \sum_{(x^n, y^n) \in A_\varepsilon^{(n)}} p(x^n) p(y^n) \\ &\geq (1-\delta) 2^{-n(H(X)+H(Y)-H(X,Y)+3\varepsilon)} \\ &\geq (1-\delta) 2^{-n(I(X;Y)+3\varepsilon)}. \end{aligned} \tag{5.28}$$

□

The above theorem suggests that we have to consider about $2^{nI(X;Y)}$ pairs before we are likely to come across a jointly typical pair.

5.2.3 Channel Coding Theorem

We are now ready to prove the other basic theorem of information theory stated by Shannon in 1948, that is the *channel coding theorem.* As previously mentioned, the remarkable result of this theorem is that, even though the channel introduce errors, the information can still be reliably sent over the channel at all rates up to channel capacity. Shannon's key idea is to sequentially use the channel many times, so that the law of large number comes into effect. Shannon's outline of the proof is indeed strongly based on the concept of typical sequences and in particular on a joint typicality based decoding rule. However, the rigorous proof was given long after Shannon's initial paper. We now give the complete statement and proof of Shannon's second theorem.

Theorem (*Channel Coding Theorem)***.**
Let us define the channel capacity as follows:

$$C = \max_{p_X(x)} I(X;Y). \tag{5.29}$$

For a discrete memoryless channel a rate R is achievable *if and only if* $R < C$.

According to the definition of achievable rate, the direct implication states that, for every rate $R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error $\lambda^{(n)} \to 0$. Conversely, the reverse implication says that for any sequence of $(2^{nR}, n)$ codes with $\lambda^{(n)} \to 0$, $R \leq C$.
Let us now prove that all the rates $R < C$ are achievable (direct implication, *if*). Later we will prove that any rate exceeding C is not achievable (converse implication,*only if*).

Proof. (*Channel Coding Theorem:* *Achievability*)
Let us fix $p_X(x)$.
For any given rate R, we have to find a proper sequence of $(2^{nR}, n)$ codes. The question that arises is how to build a codebook. It may come as a surprise that Shannon suggests to take the codewords at random. Specifically, we generate a $(2^{nR}, n)$ code according to the distribution $p(x)$ by taking 2^{nR} codewords drawn according to the distribution $p(x^n) = \prod_{i=1}^{n} p(x_i)$, thus obtaining a mapping

$$g : \{1, 2, ..., 2^{nR}\} \to \mathcal{X}^n. \tag{5.30}$$

We can organize the codewords in a matrix $2^{nR} \times n$ as follows

$$\mathcal{C} = \left(\begin{array}{cccc} x_1(1) & x_2(1) & \dots & x_n(1) \\ x_1(2) & x_2(2) & \dots & x_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(2^{nR}) & x_2(2^{nR}) & \dots & x_n(2^{nR}) \end{array} \right). \tag{5.31}$$

Each element of the matrix is drawn i.i.d. $\sim p(x)$. Each row i of the matrix corresponds to the codeword $x^n(i)$.
Having defined the encoding function g, we define the correspondent decoding function f. Shannon proposed a decoding rule based on *joint typicality*. The receiver looks for a codeword that is jointly typical with the received sequence. If a unique codeword exists satisfying this property, the receiver

declares that word to be the transmitted codeword. Formally, given y^n, if the receiver finds a unique i s.t. $(y^n, x^n(i)) \in A_\varepsilon^{(n)}$, then

$$f(y^n) = i. \tag{5.32}$$

Otherwise, that is if no such i exists or if there is more than one such codeword, an error is declared and the transmission fails. Notice that joint typical decoding is suboptimal. Indeed, the optimum procedure for minimizing the probability of error is the maximum likelihood decoding. However the proposed decoding rule is easier to analyze and asymptotically optimal.

We now calculate the average probability of error over all codes generated at random according to the above described procedure, that is

$$P_e^{(n)} = \sum_{\mathcal{C}} P_e^{(n)}(\mathcal{C}) Pr(\mathcal{C}) \tag{5.33}$$

where $P_e^{(n)}(\mathcal{C})$ is the probability of error averaged over all codewords in codebook $\mathcal{C}$. Then we have[3]

$$\begin{aligned} P_e^{(n)} &= \sum_{\mathcal{C}} Pr(\mathcal{C}) \frac{1}{2^{nR}} \sum_{i=1}^{2^{nR}} \lambda_i(\mathcal{C}) \\ &= \frac{1}{2^{nR}} \sum_{i=1}^{2^{nR}} \sum_{\mathcal{C}} Pr(\mathcal{C}) \lambda_i(\mathcal{C}). \end{aligned} \tag{5.34}$$

By considering the specific code construction we adopted, it's easy to argue that λ_i does not depend on the particular index i sent. Thus, without loss of generality, we can assume $i = 1$, yielding

$$P_e^{(n)} = \sum_{\mathcal{C}} Pr(\mathcal{C}) \lambda_1(\mathcal{C}). \tag{5.35}$$

If Y^n is the result of sending $X^n(i)$ over the channel[4], we define the event E_i as the event that the i-th codeword and the received one are jointly typical, that is

$$E_i = \{(X^n(i), Y^n) \in A_\varepsilon^{(n)}\}, \quad i \in \{1, 2, ..., 2^{nR}\}. \tag{5.36}$$

[3] We precise that there is a slight abuse of notation, since $P_e^{(n)}(\mathcal{C})$ in (5.33) corresponds to $P_e^{(n)}$ in (5.20), while $P_e^{(n)}$ in (5.33) denotes the probability of an error averaged over all the codes. Similarly, $\lambda_i(\mathcal{C})$ corresponds to λ_i where again the dependence on the codebook is made explicit.

[4] Both $X^n(i)$ and Y^n are random since we are not conditioning to a particular code. We are interested in the average on C.

Since we assumed $i = 1$, we can define the *error event* $\mathcal{E}$ as the union of all the possible types of error which may occur during the decoding procedure (jointly typical decoding):

$$\mathcal{E} = E_1^c \cup E_2 \cup E_3 \cup ... \cup E_{2^{nR}}, \tag{5.37}$$

where the event E_1^c occurs when the transmitted codeword and the received one are not jointly typical, while the other events refer to the possibility that a wrong codeword (different from the transmitted one) is jointly typical with Y^n (the received sequence). Hence: $Pr(\mathcal{E}) = Pr(E_1^c \cup E_2 \cup E_3 \cup ... \cup E_{2^{nR}})$. We notice that the transmitted codeword and the received sequence must be jointly typical, since they are probabilistically linked through the channel. Hence, by bounding the probability of the union in (5.37) with the sum of the probabilities, from the first and the fourth point of the joint AEP theorem we obtain

$$\begin{aligned}
Pr(\mathcal{E}) &\leq Pr(E_1^c) + \sum_{i=2}^{2^{nR}} Pr(E_i) \\
&\leq \delta + \sum_{i=1}^{2^{nR}} 2^{-n(I(X;Y)-3\varepsilon)}, \\
&\leq \delta + (2^{nR} - 1)2^{-n(I(X;Y)-3\varepsilon)}, \\
&\leq \delta + 2^{nR}2^{-n(I(X;Y)-3\varepsilon)}, \\
&\leq \delta + 2^{-n(I(X;Y)-R-3\varepsilon)} \\
&= \delta',
\end{aligned} \tag{5.38}$$

where δ' can be made arbitrarily small for $n \to \infty$ if $R < I(X;Y)$. The intuitive meaning of the above derivation is the following: since for any codeword, different from the transmitted one, the probability to be jointly typical with the received sequence is approximately $2^{-nI(X;Y)}$, we can use at most $2^{nI(X;Y)}$ codewords in order to keep the error probability arbitrarily small for large enough n. In other words, if we have not too many codewords ($R < I$), with high (arbitrarily close to 1) probability there is no other codeword that can be confused with the transmitted one.

At the beginning of the proof, we fixed $p_X(x)$ which determines the value of $I(X;Y)$. Actually $p_X(x)$ is the ultimate degree of freedom we can exploit in order to obtain the smallest $Pr(\mathcal{E})$ for the given rate R. As a consequence, it is easy to argue that $Pr(\mathcal{E})$ can be made arbitrarily small (for large n) if the rate R is less than the maximum of mutual information, that is

$$C = \max_{p_X(x)} I(X;Y). \tag{5.39}$$

To conclude the proof we need a further step. In fact, the achievability definition is given in terms of the maximal probability of error $\lambda_{max}^{(n)}$, while up to now we have dealt with the average probability of error. We now show that

$$P_e^{(n)} \to 0 \quad \Rightarrow \quad \exists \mathcal{C} \quad \text{s.t} \quad \lambda_{max}^{(n)} \to 0. \tag{5.40}$$

Since $P_e^{(n)} = Pr(\mathcal{E}) < \delta'$, there exists at least one code $\mathcal{C}$ (actually more than one) such that $P_e^{(n)}(\mathcal{C}) < \delta'$. Name it $\mathcal{C}^*$. Let us list the probabilities of error λ_i of the code $\mathcal{C}^*$ in increasing order:

$$\lambda_1, \lambda_2, ..., \lambda_{2^{nR}}.$$

Now, we throw away the upper half of the codewords in $\mathcal{C}^*$, thus generating a new code $\mathcal{C}^\star$ with half codewords. Being the average probability of error for the code $\mathcal{C}^*$ lower than δ' we deduce that

$$\lambda_{\frac{2^{nR}}{2}} < 2\delta'. \tag{5.41}$$

(If it were not so, it is easy to argue that $P_e^{(n)}(\mathcal{C})$ would be greater than δ'.) But $\lambda_{2^{nR}/2}$ is the maximal probability of error for the code $\mathcal{C}^\star$, which then is arbitrarily small (tends to zero as $n \to \infty$).
What about the rate of $\mathcal{C}^\star$? Throwing out half the codewords reduces the rate from R to $R - \frac{1}{n}$ $(= \log(2^{nR-1})/n)$. This reduction is negligible for large n. Then, for large n, we have found a code having rate R and whose $\lambda_{max}^{(n)}$ tends to zero. This concludes the proof that any rate below C is achievable.
□

Some considerations can be made regarding the proof: similarly to the source coding theorem, Shannon does not provide any usable way to construct the codes. The construction procedure used in the proof is highly impractical for many reasons. Firstly, Shannon's approach is asymptotical: both the number of codewords, 2^{nR}, and the length, n, have to go to infinity. Secondly, but not least, Shannon suggests to generate the code at random; accordingly, we should write down all the codewords in the matrix $\mathcal{C}$ (see (5.31)) and moreover transmit the matrix to the receiver. It is easy to guess that, for large values of n, this scheme requires (storage and transmission) resources out of any proportion. In fact, without some structure in the code it is not possible to decode. Only structured codes (i.e. codes generated according to a rule) are easy to encode and decode in practice.

Now we must show that it is not possible to 'do better' than C (converse). Before giving the proof we need to introduce two lemmas of general validity.

Lemma (Fano's inequality)**.** *Let X and Y be two dependent sources and let g be any deterministic reconstruction function s.t. $\hat{X} = g(Y)$. The following upper bound on the remained uncertainty (or equivocation) about X given Y holds:*

$$\begin{aligned} H(X|Y) &\leq h(P_e) + P_e \log(|\mathcal{X}| - 1) \\ &\leq 1 - P_e \log(|\mathcal{X}| - 1), \end{aligned} \tag{5.42}$$

where $P_e = Pr(\hat{X} \neq X)$.

Proof. We introduce an error random variable

$$E = \begin{cases} 1 & \text{if } \hat{x} \neq x \quad \text{(with probability } P_e) \\ 0 & \text{if } \hat{x} = x \quad \text{(with probability } 1 - P_e). \end{cases} \tag{5.43}$$

By using the chain rule we can expand $H(E, X|Y)$ in two different ways:

$$\begin{aligned} H(X, E|Y) &= H(X|Y) + H(E|X, Y) && (5.44) \\ &= H(E|Y) + H(X|E, Y). && (5.45) \end{aligned}$$

It's easy to see that $H(E|X, Y) = 0$ while $H(E|Y) < H(E) = h(P_e)$. As to $H(X|E, Y)$, by expliciting the sum on E we have

$$H(X|E, Y) = (1 - P_e)H(X|0, Y) + P_e H(X|1, Y). \tag{5.46}$$

Relation (5.46) can be simplified by observing that, when $E = 0$, there is no uncertainty on the value of X (that is, being $\hat{x} = x$, $H(X|0, Y) = 0$) while, when $E = 1$, the estimation of X is not correct (being $\hat{x} \neq x$). Using the bound on the maximum entropy yields $H(X|1, Y) \leq \log(|\mathcal{X}| - 1)$. Then, the sum in (5.46) can be written as:

$$H(X|E, Y) =\leq P_e \log(|\mathcal{X}| - 1). \tag{5.47}$$

By expliciting $H(X|Y)$ from equality (5.44)-(5.45) we eventually have

$$\begin{aligned} H(X|Y) &\leq h(P_e) + P_e \log(|\mathcal{X}| - 1) \\ &\leq 1 - P_e \log(|\mathcal{X}| - 1), \end{aligned} \tag{5.48}$$

which is the desired relation.
The second inequality provides a weaker upper bound which however allows to avoid the evaluation of the binary entropy $h(P_e)$. □

Fano's inequality is useful whenever we know a random variable Y and we wish to guess the value of a correlated random variable X. It relates the probability of error in guessing the random variable X, i.e. P_e, to the conditional entropy $H(X|Y)$.
It's interesting to note that Fano's inequality can also be seen as a lower bound on P_e. Looking at X and Y as the input and the output of a channel and looking at g as the decoding function, P_e corresponds to the probability of a decoding error[5].

Lemma. *Let us consider a discrete memoryless channel (DMC) with input and output sources X and Y. By referring to the extended channel we have*

$$I(X^n; Y^n) \leq nC. \tag{5.49}$$

Proof.

$$\begin{aligned} I(X^n; Y^n) &= H(Y^n) - H(Y^n|X^n) \\ &\overset{(a)}{=} H(Y^n) - \sum_i H(Y_i|Y_{i-1}, ..., Y_1, X^n) \\ &\overset{(b)}{=} H(Y^n) - \sum_i H(Y_i|X_i) \\ &\overset{(c)}{\leq} \sum_i H(Y_i) - \sum_i H(Y_i|X_i) \\ &= I(Y_i; X_i) \leq nC, \end{aligned} \tag{5.50}$$

where (a) derives from the application of the generalized chain rule and (b) follows from the memoryless (and no feedback) assumption. Since conditioning reduces uncertainty, $H(Y^n) \leq \sum_i H(Y_i)$, we have relation (c). We stress that the output symbols Y_i do not need to be independent, that is generally $p(y_i|y_{i-1}, ..., y_1) \neq p(y_i)$. Since C is defined as the maximal mutual information over $p(x)$ the last inequality clearly holds. □

The above lemma shows that using the channel many times does not increase the transmission rate.

[5] For sake of clarity, we point out that Fano's inequality holds even in the more general case in which the function $g(Y)$ is random, that is for any estimator $\hat{X}$ such that $X \rightarrow Y \rightarrow \hat{X}$.

Remark: the lemma holds also for non DM channels, but this extension is out of the scope of these notes.

We have now the necessary tools to prove the converse of the channel coding theorem.

Proof. (*Channel Coding Theorem: Converse*)
We show that any sequence of (2^{nR},n) codes with $\lambda_{max}^{(n)} \to 0$ must have $R \leq C$; equivalently, if $R > C$ then $P_e^{(n)}$ cannot tend to 0 (thus implying that $\lambda_{max}^{(n)}$ does not tend to 0).
Given the index set $\{1, 2, ..., 2^{nR}\}$, a fixed encoding function which associates to a index (message) W a codeword $X^n(W)$, and a fixed decoding rule $g(\cdot)$ such that $\hat{W} = g(Y^n)$, we have

$$W \to X^n(W) \to Y^n \to \hat{W}. \tag{5.51}$$

In (5.51), Y^n takes the role of the observation, W the role of the index we have to estimate and $Pr(\hat{W} \neq W) = P_e^{(n)} = \frac{1}{2^{nR}} \sum_i \lambda_i$. The random variable W corresponds to a uniform source, since the indexes are drawn in an equiprobable manner, thus the entropy has the expression $H(W) = \log(2^{nR})$. By using the definition of the mutual information we have

$$nR = H(W) = I(W; Y^n) + H(W|Y^n). \tag{5.52}$$

Since the channel directly acts on X^n, we deduce that $p(y^n|x^n, w) = p(y^n|x^n)$, that is $W \to X^n \to Y^n$. Then, according to the properties of the Markov chains and in particular to DPI, from (5.52) it follows that

$$nR \leq I(X^n; Y^n) + H(W|Y^n). \tag{5.53}$$

By exploiting the lemmas proved above, from (5.53) we get

$$\begin{aligned} nR &\leq I(X^n; Y^n) + 1 + P_e^{(n)} \log(2^{nR} - 1) \\ &< nC + 1 + P_e^{(n)} nR. \end{aligned} \tag{5.54}$$

Dividing by n yields:

$$R < C + \frac{1}{n} + P_e^{(n)} R. \tag{5.55}$$

It follows that if $n \to \infty$ and $P_e^{(n)} \to 0$ then $R < C + \varepsilon$ for any arbitrarily small ε, i.e. $R \leq C$.
According to the direct channel coding theorem, n must tend to infinity so

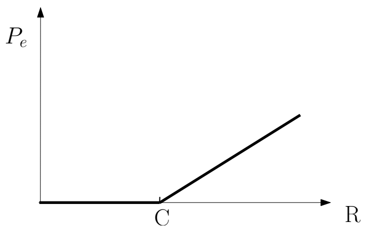


Figure 5.10: Asymptotic lower bound on P_e by varying R.

that $P_e^{(n)}$ can be made arbitrarily small. Therefore, if we want $P_e^{(n)} \to 0$ it's necessary that the rate R stays below capacity. This fact proves that $R < C$ is also a necessary condition for a rate R to be achievable.
From (5.55) there is another possible way through which we can show that if $R > C$ then $P_e^{(n)} \nrightarrow 0$. Let us rewrite (5.55) as follows

$$P_e^{(n)} \geq 1 - \frac{C}{R} - \frac{1}{nR}. \tag{5.56}$$

Joining this condition with the positivity of $P_e^{(n)}$ produces the asymptotical lower bound on P_e depicted in Figure 5.10. It's easy to see that if $R > C$ the probability of error is bounded away from 0 for large n. As a consequence, we cannot achieve an arbitrarily low probability of error at rates above capacity.

□

5.2.4 Channel Coding in practice

The essence of the channel coding theorem is that, as long as $R < C$, it is possible to send information without affecting the reliability of the transmission. Hence, the noisiness of the channel does not limit the reliability of the transmission but only its rate. Moreover, Shannon proves that choosing the codes at random is asymptotically the best choice whatever the channel is. However, it is easy to deduce that for finite n the knowledge of the channel may help to choose a better code.
The problems we have to face with in practice are many. Hereinafter, we review the most common channels in order to compute the channel capacity C.

Evaluation of channel capacity

In order to evaluate the channel capacity of a given channel we have to solve the maximization

$$C = \max_{p(x)} I(X;Y), \tag{5.57}$$

for a given $p(y|x)$ and subject to the constraints on $p(x)$,

$$\begin{cases} p(x) \in [0,1] & \forall x \\ \sum_x p(x) = 1. \end{cases} \tag{5.58}$$

It's possible to prove that since $p(y|x)$ is fixed by the channel, the mutual information is a *concave function* of $p(x)$. Hence, a maximum for $I(X;Y)$ exists and is unique. However, being the objective function a nonlinear function, solving (5.57) is not easy and requires using methods of numerical optimization. There are only some simple channels, already introduced at the beginning of the chapter, for which it is possible to determine C analytically.

- *Noisy typewriter*

In this channel if we know the input symbol we have two possible outputs (the same or the subsequent symbol) with a probability 1/2 for each. Then, $H(Y|X) = 1$ and $\max I(X;Y) = \max(H(Y) - H(Y|X))) = \max(H(Y) - 1)$. The maximum of the entropy of the output source, which is $\log|\mathcal{Y}|$, can be achieved by using $p(x)$ distributed uniformly over all the inputs. Since the input and the output alphabet coincide, we have

$$C = \log|\mathcal{Y}| - 1 = \log|\mathcal{X}| - 1. \tag{5.59}$$

We deduce that, due to the action of the channel, we loose 1 information bit. Equivalently, the maximum rate of transmission is $C = \log\frac{|\mathcal{X}|}{2}$. This suggests that the intuitive idea of considering half symbols we proposed at the beginning of Section 5.2 is an optimum choice. It may come as a paradox that the value C is obtained by considering the inputs equally likely, but this is not necessarily the way according to which we have to take the inputs if we want to transmit at rate C. In fact, in this particular case, taking only non consecutive inputs permits to send information through the channel at the maximum rate C, without having to send n to infinity. This is not a contradiction; Shannon proposes a conceptually simple encoding and decoding scheme, this does not preclude the existence of better schemes, especially for finite n. What is certain is that the transmission rate cannot go beyond C.

- *BSC*

Even for this channel the maximization of the mutual information is straightforward, since we can easily compute the probability distribution $p(x)$ which maximizes $H(Y)$. As we already know from the analysis in Section 5.2.1, $C = \max(H(Y) - h(\varepsilon)) = 1 - h(\varepsilon)$, which is achieved when the input distribution is uniform.

- *BEC*

For the binary erasure channel (Figure 5.7) the evaluation of the capacity is a little bit more complex. Since $H(Y|X)$ is a characteristic of the channel and does not depend on the probability of the input, we can write

$$\begin{aligned} C &= \max_{p(x)}(H(Y) - H(Y|X)) \\ &= \max_{p(x)} H(Y) - h(\alpha). \end{aligned} \tag{5.60}$$

For a generic value of α the absolute maximum value for $H(Y)$ ($\log|\mathcal{Y}| = \log 3$) cannot be achieved for any choice of the input distribution. Then, we have to explicitly solve the maximization problem. Let $p_X(0) = \pi$ and $p_X(1) = 1 - \pi$. There are two ways for the evaluation of π. According to the first method, from the output distribution given by the triplet $p_Y(y) = (\pi(1-\alpha), \alpha, (1-\pi)(1-\alpha))$ we calculate the entropy $H(Y)$ and later maximize on π. The other method exploits the grouping property, yielding

$$\begin{aligned} H(Y) =& H_3(\pi(1-\alpha), \alpha, (1-\pi)(1-\alpha)) \\ =& H_2(\alpha, (1-\alpha)) + (1-\alpha)H_2(\pi, 1-\pi) = h(\alpha) + (1-\alpha)h(\pi). \end{aligned} \tag{5.61}$$

The maximum of the above expression is obtained when $h(\pi) = 1$, and then for $\pi = 1/2$. It follows that $C = h(\alpha) + (1-\alpha) - h(\alpha) = 1 - \alpha$. The result is expected since the BEC channel is nothing else that a noiseless binary channel which breaks down with a probability α; then, C can be obtained substracting to 1 the fraction of time the channel remains inoperative.

Construction of the codes

The channel coding theorem promises the existence of block codes that allow to transmit information at rates below capacity with arbitrarily small probability of error if the block length is large enough. The greatest problem of channel coding is to find codes which allows in practice to transmit at rate close to C. Ever since the appearance of Shannon's paper, people have searched for such codes. In addition, usable codes should be "simple", so that they could be encoded and decoded easily. If we generated the codewords at random, according to Shannon's scheme, we would have to list all the codewords and send them to the receiver, requiring a huge amount of memory. Furthermore, we need a way to associate the messages we have to transmit and the codewords. Besides, since the code must be invertible, the codewords have to be distinct among themselves. Shannon overcomes this problem considering an asymptotical situation. Sending n to infinity is also what makes possible to use the jointly typical decoding as decoding rule at the receiver side. Such decoding scheme requires the receiver to check all the sequences which may have been sent in order to make the decision on the transmitted codeword. However, even if we consider a minimum distance algorithm it may require up to 2^{nR} evaluations.

Chapter 6

Continuous Sources and Gaussian Channel

In this chapter, we deal with continuous sources. By following the same steps of the analysis developed for the discrete case we highlight the conceptual differences the continuity assumption leads to.

6.1 Differential Entropy

Let X be a random variable taking values in $\mathbb{R}$ characterized by a *probability density function* $f_X(x)$ [1].

Definition. The *differential entropy h(X)* is defined as

$$h(X) = -\int_{\mathbb{R}} f_X(x) \log f_X(x) dx. \tag{6.1}$$

The lower case letter h is used in place of the capital letter H denoting the entropy in the discrete case.
It can be shown that the differential entropy represents a valid measure for the information carried by a continuous random variable: indeed, if $h(X)$ grows the prior uncertainty about the value of X increases. However, some of the intuitiveness of the entropy is lost. The main reason for this is that now the differential entropy can take negative values: this happens for instance when we compute the entropy of a random variable with a uniform distribution in a continuous range $[0, a]$ where $a < 1$ (in this case in fact $h(X) = \log a < 0$).

[1] In the continuous case we refer to pdf instead of pmf.

The quantities related to the differential entropy, like the joint and conditional entropy, mutual information, and divergence, can be defined in the same way as for the discrete case[2] and most of their properties proved likewise.

6.2 AEP for Continuous Sources

We now revisit the AEP theorem which still holds for continuous memoryless sources. The proof is omitted since it is very similar to that of the AEP theorem for the discrete case. A major difference which must be remarked is the fact that, dealing with continuous alphabets, the typical sequences can not be counted or listed. Then, in the continuous case, we refer to the *volume* occupied by the set of typical sequences, which in turn is about $2^{nh(X)}$. By looking at this approximated value for the volume, it is worth noting that a negative differential entropy corresponds to a small (but always positive) volume occupied by the set of typical sequences, not implying any contradiction. Furthermore, as the intuition suggests, a low uncertainty about the sequences is associated to a small volume occupied by the typical sequences. For completeness, we give the formal definition of the volume. Assuming that a set S is measurable sets according to Riemann or Lebesgue measures, the volume of S is defined as

$$Vol(S) = \underset{S}{\int \cdots \int} d\vec{x}, \tag{6.2}$$

where $\vec{x} = (x_1, x_2, ..., x_n)$. We can now state the AEP theorem.

Theorem. *(AEP: continuous case)*
Given a CMS[3], $X \sim f_X(x)$ and defined the set $A_\varepsilon^{(n)}$ as follows:

$$A_\varepsilon^{(n)} = \left\{ x^n : \left| -\frac{1}{n} \log f_{X_1,X_2,...,X_n}(x_1, x_2, ..., x_n) - h(X) \right| < \varepsilon \right\}, \tag{6.3}$$

we have:

1. $\forall \delta > 0, \forall \varepsilon > 0, n$ large, $\quad \Pr\{A_\varepsilon^{(n)}\} \geq 1 - \delta$;
2. $\forall \varepsilon, \forall n, \quad Vol(A_\varepsilon^{(n)}) \leq 2^{n(h(X)+\varepsilon)}$;
3. $\forall \delta > 0, \forall \varepsilon > 0, n$ large, $\quad Vol(A_\varepsilon^{(n)}) \geq (1 - \delta) 2^{n(h(X)-\varepsilon)}$.

[2] by paying attention to replace the sum with the integral.

[3] Continuous memoryless source.

Observation.
From the AEP theorem we know that the differential entropy is directly related to the volume occupied by the typical sequences. The following intuitive properties hold:

1. $h(X) = h(X - \mu)$,
2. $h(X) \neq h(\alpha X)$,

where α is any scale factor.
Equality 1 follows from the translation invariance of the volume. We stress that the same relation holds for the entropy in the discrete case. We leave the proof as exercise.
Inequality 2 is due to the fact that scaling changes the volume. This property contrasts with the discrete case for which it's easy to deduce that $H(X) = H(\alpha X)$. Let us prove such inequality. It is known that

$$X \sim f_X(x) \quad \text{implies} \quad Y = \alpha X \sim \frac{1}{|\alpha|} f_X\left(\frac{y}{\alpha}\right). \tag{6.4}$$

Hence ($\alpha > 0$),

$$h(Y) \;=\; -\int \frac{1}{\alpha} f_X\left(\frac{y}{\alpha}\right) \log\left[\frac{1}{\alpha} f_X\left(\frac{y}{\alpha}\right)\right] dy. \tag{6.5}$$

By performing a variable change from y to x (modifying the differential and the support of the integral) we get

$$\begin{aligned} &= -\int f_X(x) \log \frac{1}{\alpha} dx - \int f_X(x) \log f_X(x) dx \\ &= h(X) + \log \alpha \neq h(X). \end{aligned} \tag{6.6}$$

Observe that the additional term $\log \alpha$ corresponds to the (volume) scaling n-dimensional factor being $Vol \approx 2^{n(h(X)+\log \alpha)} = \alpha^n 2^{nh(X)}$.

6.3 Gaussian Sources

The calculation of the differential entropy is somewhat problematic and actually impossible for a generic probability density function $f_X(x)$. A remarkable case for which the computation is particularly easy is the case of a Gaussian density function.

The probability density function of a Gaussian random variable X is:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{6.7}$$

where μ is the expected value (expectation) of the distribution and σ the standard deviation (i.e. σ^2 is the variance). The Gaussian (or normal) distribution in (6.7) is often denoted by $\mathcal{N}(\mu, \sigma^2)$.

Let us compute the differential entropy of the Gaussian distributed random variable X:

$$\begin{aligned}
h(X) &= -\int_{\mathbb{R}} f_X(x) \cdot \log f_X(x) \\
&= -\int_{\mathbb{R}} \mathcal{N}(\mu, \sigma^2) \log \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} dx \\
&= \log \sqrt{2\pi\sigma^2} + \log e \int_{\mathbb{R}} \mathcal{N}(\mu, \sigma^2) \frac{(x-\mu)^2}{2\sigma^2} dx \\
&= \frac{1}{2} \log 2\pi\sigma^2 + \frac{\log e}{2\sigma^2} \int_{\mathbb{R}} (x-\mu)^2 \cdot \mathcal{N}(\mu, \sigma^2) dx \\
&\overset{(a)}{=} \frac{1}{2} \log 2\pi\sigma^2 + \frac{\log e}{2} \\
&= \frac{1}{2} \log 2\pi e\sigma^2. \qquad (6.8)
\end{aligned}$$

where in (a) we exploit the definition of the variance: $\sigma^2 = E[(x-\mu)^2] = \int_{\mathbb{R}} (x-\mu)^2 \cdot f_X(x) dx$.

Let us now consider the general case of n jointly Gaussian random variables forming a Gaussian vector $\vec{X} = X_1, ..., X_n$. We want to evaluate the differential entropy $h(X_1, X_2, ..., X_n)$. A Gaussian vector $\vec{X}$ is distributed according to a *multivariate Gaussian density function* which has the expression

$$f_{\vec{X}}(\vec{x}) = \frac{1}{\sqrt{(2\pi)^n |C|}} e^{-\frac{(\vec{x}-\vec{\mu}) C^{-1} (\vec{x}-\vec{\mu})^T}{2}}, \tag{6.9}$$

where $\vec{\mu}$ is the vector of the expected values μ_i of the random variables X_i $(i = 1, ..., n)$, and C is the covariance matrix. In the sequel we will use the compact notation C_{ij} to denote the element (i, j) of the covariance matrix C, i.e. $C_{ij} = cov(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)]$. The Gaussian (normal) density function of a random vector $\vec{X}$ with mean $\vec{\mu}$ and covariance C is commonly referred to as $\mathcal{N}(\vec{\mu}, C)$.

Note that if C is a *diagonal* matrix, that is the n r.v.'s are independent, we

obtain the product of n one-dimensional Gaussian pdf's $\mathcal{N}(\mu_i, \sigma_i^2)$.
Let us now compute the entropy of the random Gaussian vector $\vec{X}$ or equivalently the joint entropy of the n random variables X_i:

$$\begin{aligned}
h(\vec{X}) &= -\int \cdots \int_{\mathbb{R}^n} \mathcal{N}(\vec{\mu}, C) \cdot \log \frac{e^{\frac{-(\vec{x}-\vec{\mu})C^{-1}(\vec{x}-\vec{\mu})^T}{2}}}{\sqrt{(2\pi)^n \, |C|}} \\
&\overset{4}{=} \log \sqrt{(2\pi)^n \, |C|} + \log e \int \cdots \int_{\mathbb{R}^n} \mathcal{N}(\vec{\mu}, C) \frac{(\vec{x}-\vec{\mu})C^{-1}(\vec{x}-\vec{\mu})^T}{2} d\vec{x} \\
&= \frac{1}{2} \log(2\pi)^n \, |C| + \frac{\log e}{2} \int \cdots \int_{\mathbb{R}^n} \mathcal{N}(\vec{\mu}, C) \cdot \sum_i \sum_j (x_i - \mu_i)(C^{-1})_{ij}(x_j - \mu_j) d\vec{x} \\
&= \frac{1}{2} \log(2\pi)^n \, |C| + \frac{1}{2} \log e \cdot \sum_i \sum_j (C^{-1})_{ij} \int \cdots \int_{\mathbb{R}} \mathcal{N}(\vec{\mu}, C)(x_i - \mu_i)(x_j - \mu_j) d\vec{x} \\
&= \frac{1}{2} \log(2\pi)^n \, |C| + \frac{1}{2} \log e \cdot \sum_i \sum_j (C^{-1})_{ij} \cdot C_{ij}. \qquad (6.10)
\end{aligned}$$

By exploiting the symmetry of the covariance matrix the inner sum in the last equation of 6.10 can be rewritten as $\sum_j (C^{-1})_{ij} \cdot C_{ji}$, which is nothing else then the (i, i) element of the matrix product $C^{-1} \cdot C$ (i.e. the identity matrix). Then, going on from (6.10) we get:

$$\begin{aligned}
h(\vec{X}) &= \frac{1}{2} \log(2\pi)^n \, |C| + \frac{1}{2} \log e \cdot \sum_{i=1}^{n} I_{ij} \\
&= \frac{1}{2} \log[(2\pi e)^n \, |C|]. \qquad (6.11)
\end{aligned}$$

We have found the expression of the entropy of a n-length vector of jointly Gaussian random variables. Setting $n = 1$ yields the entropy of a Gaussian random variable, that is

$$h(X) = \frac{1}{2} \log 2\pi e \sigma^2. \qquad (6.12)$$

As expected, if the n random variables are independent (C is diagonal and then $|C| = \prod_i \sigma_i^2$) we have $h(\vec{X}) = \sum_i h(X_i) = \sum_i \frac{1}{2} \log 2\pi e \sigma_i^2$.

We now prove that, among all the possible continuous distributions with the same variance, the Gaussian distribution is the one that has the largest entropy.

[4] We make use of the unitary sum property of the density function: $\int \cdots \int_{R^n} \mathcal{N}(\vec{\mu}, C) = 1$.

Property. Let $f(x)$ be a Gaussian density function with variance σ^2 and let $g(x)$ be any other density function having the same variance. Then

$$h(f) \geq h(g)^5. \tag{6.13}$$

Proof.

$$\begin{aligned} 0 \leq \mathcal{D}(g(x)||f(x)) &= \int g(x) \log \frac{g(x)}{f(x)} dx \\ &= \int g(x) \log g(x) - \int g(x) \log f(x) dx \\ &\overset{(a)}{=} -h(g) - \int g(x) \log \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} dx \\ &= -h(g) - \log \frac{1}{\sqrt{2\pi\sigma^2}} + \log e \cdot \int g(x) \frac{x^2}{2\sigma^2} dx \\ &= -h(g) - \log \frac{1}{\sqrt{2\pi\sigma^2}} + \frac{1}{2\sigma^2} \log e \cdot \int x^2 g(x) dx. \\ &= -h(g) + \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2} \log e, \\ &= -h(g) + h(f), \end{aligned} \tag{6.14}$$

where in (a), without any loss of generality, we considered a zero mean density function (as we will see, the differential entropy does not depend on the mean value). From (6.14), we can easily obtain the desired relation. □

Note: the previous property can be used to give a meaningful justification to the Gaussian assumption often made in noise characterization. According to the property, this is a *worst case* assumption since it corresponds to consider the situation of maximum a priori uncertainty. In other words, making Gaussian assumption corresponds to apply *the principle of maximum entropy.*

6.4 Gaussian Channel (AWGN)

In Chapter 5 we introduced the discrete communication channel and channel coding and analyzed the most common discrete channel models. For each

[5] Here, we use a slight different notation to indicate the differential entropy in order to make explicit the density function according to which the variable is generated.

model (BSC, BEC,...), we have supposed that the channel is characterized by a transition matrix, thus viewing it as a black box. The analysis of continuous sources allows us to directly consider the analog (physical) channel inside which the continuous modulated waveform are transmitted. In this way, we can configure the connection by designing the most appropriate modulation for a given physical channel. From the knowledge of digital modulation theory we know that the error probability of a link is related to the bandwidth and the power of the transmitted signal.

In this chapter we formally describe the Gaussian channel [6] by specifying the relation between the input X and the output Y of the channel. The AWGN channel is characterized by an additive relation between input and output; the output Y is obtained by adding to X a white Gaussian noise Z, that is

$$Y = X + Z, \qquad Z \sim \mathcal{N}(0, \sigma_z^2). \tag{6.15}$$

Being the added noise white, the channel is stationary and memoryless. This channel is a model for a number of common communication channels, such as the telephone channel and satellite links.

Without any limitation on the input, we argue that the capacity of the Gaussian channel is infinite and we can obtain a perfect (with no error) transmission, as if the noise variance were zero. However, it is quite reasonable to assume that the power of the input signal is constrained. In particular we require that using the channel n times yields a transmitted power less then P_X (or σ_x^2), i.e.

$$\frac{1}{n}\sum_i x_i^2 \leq P_X. \tag{6.16}$$

Given the constraint in (6.16), we are interested to determine the maximum rate at which transmission is possible through the channel. It's proper to stress that, strictly speaking, we still have to demonstrate that the channel capacity concept holds for the continuous case. Before rigorously formalizing these concepts we empirically show the basic ideas behind channel coding for transmission over an AWGN channel.

6.4.1 The Coding problem: a qualitative analysis

As done for the discrete case, we introduce the channel coding problem through a qualitative analysis.

Consider the n-th extension of the channel. Working with a continuous alphabet causes an infinite number of possible sequences x^n that can be

[6] We still assume a discrete-time channel.

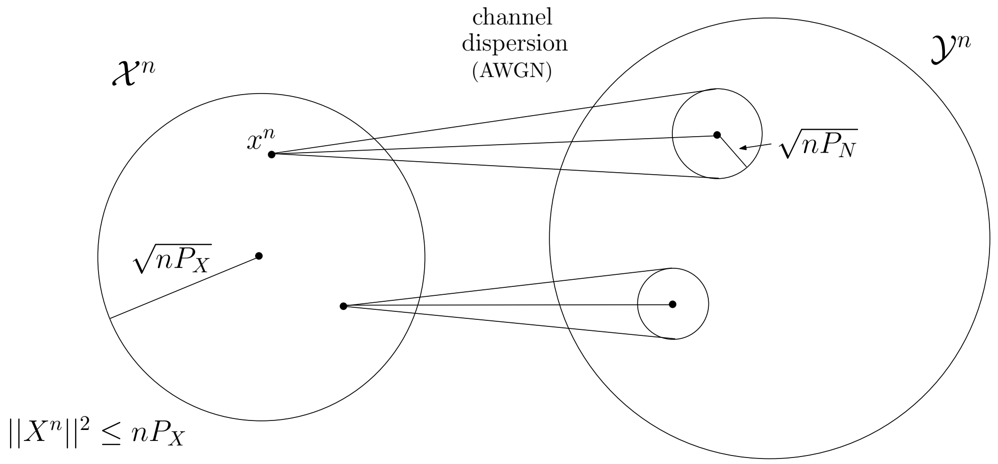


Figure 6.1: Representation of the n-the extended Gaussian channel.

transmitted over the channel through n uses. Due to the noise added by the channel during the transmission, for any input sequence there are in turn an infinite number of possible outputs. The power constraint $||x^n|| \leq \sqrt{nP_X}$ allows to say that all the possible inputs lie in a n-dimensional hypersphere (in $\mathbb{R}^n$) of radius $\sqrt{nP_X}$ (see Figure 6.1). What we want to determine is the maximum number of sequences that can be *reliably transmitted* over the channel (error-free transmission). Looking at the figure, we see that without limitation imposed on the power of the input signal we could reliably transmit an infinite number of sequences (being the radius of the sphere unbounded), despite the dispersion caused by the noise.

In order to find the maximum number of reliably transmissible sequences we can compute the maximum number of disjoint sets we can dispose in the output space ($\mathcal{Y}^n$) (Figure 6.1). Each sequence y^n in the set of output sequences is obtained by the sum $x^n + z^n$, where x^n is the corresponding input sequence and z^n is a Gaussian noise vector. Each coefficient z_i represents the noise relative to the the i-th use of the channel ($Z_i \sim \mathcal{N}(0, \sigma_z^n)$, being the Z_i i.i.d.). The random output vector $Y^n = x^n + Z^n$ has a Gaussian distribution with mean x^n and the same variance of the noise, i.e. $\sigma_z^2 = P_N$. Therefore, it's correct to represent the output set centered on the input sequence. Besides, if n is sufficiently large, we can affirm that with high probability the output points lie on the boundary of the $\sqrt{nP_N}$-radius hypersphere since for the Law of Large Numbers

$$||(x^n + Z^n) - x^n||^2 = ||Z^n||^2 = \sum_i Z_i^2 \longrightarrow nP_N \quad \text{as } n \to \infty. \tag{6.17}$$

We now evaluate the volume of the n-dimensional hypersphere containing approximately, and with high probability, all the "typical" output sequences. In order to do this, we observe that a generic point of the output space can be denoted as $X^n + Z^n$. The total power, for large n, with high probability, is

$$\begin{aligned} ||X^n + Z^n||^2 &= ||X^n||^2 + ||Z^n||^2 + 2 < X^n \cdot Z^n > \\ &= \sum_i X_i^2 + \sum_i Z_i^2 + 2 \sum_i X_i Z_i \qquad (6.18) \\ &\leq nP_X + nP_N. \end{aligned}$$

where in equality (6.18) we exploited the independence of the signal from the noise. Then, the received vectors lie inside a sphere of radius $\sqrt{n(P_X + P_N)}$. Being the volume directly proportional to the n-th power of the radius with a proportionality constant a_n, the maximum number of non-overlapping (non-intersecting) spheres which is possible to arrange in this volume is bounded by

$$\frac{a_n(n(P_X + P_N))^{n/2}}{a_n(nP_N)^{n/2}} = \left(\frac{P_N + P_X}{P_N}\right)^{n/2}. \qquad (6.19)$$

Then, the number of bits that can be reliably transmitted for each use of the channel is at most

$$\frac{1}{2} \log \left(1 + \frac{P_X}{P_N}\right). \qquad (6.20)$$

The above arguments tell us that we cannot hope to send information at rate larger then the value in (6.20) with no error. In the next section we will rigorously prove that as $n \to \infty$ we can do almost as well as this.

6.4.2 Coding Theorems for the Gaussian Channel

Even for the continuous case it's possible to refer to the same definitions of code, error probability and achievable rate given in Section 5.2.2 for the discrete case. Before stating and proving the channel coding theorem for the Gaussian case, we define the *jointly typical set* and enunciate the joint AEP theorem.

Definition. Let X and Y be two continuous sources with probability density function $f_X(x)$ and $f_Y(y)$ respectively and joint pdf $f_{XY}(x, y)$. The *jointly typical set* $A_\varepsilon^{(n)}$ is defined as:

$$\begin{aligned} A_\varepsilon^{(n)} = \{&(x^n, y^n) \in \mathcal{X}^n \times Y^n : \\ &\left| -\frac{1}{n} \log f_X(x^n) - h(X) \right| < \varepsilon, \left| -\frac{1}{n} \log f_Y(y^n) - h(Y) \right| < \varepsilon, \\ &\left| -\frac{1}{n} \log f_{XY}(x^n, y^n) - h(X,Y) \right| < \varepsilon \bigg\}. \end{aligned} \tag{6.21}$$

Using the above definition we can state the following theorem:

Theorem (*Joint AEP: continuous case)*.

1. $\forall \delta > 0, \forall \varepsilon > 0, n$ large, $\quad \Pr\{A_\varepsilon^{(n)}\} \geq 1 - \delta$;
2. $\forall \varepsilon, \quad Vol(A_\varepsilon^{(n)}) \leq 2^{n(h(X,Y)+\varepsilon)} \quad \forall n$;
3. $\forall \delta > 0, \forall \varepsilon > 0, n$ large, $\quad Vol(A_\varepsilon^{(n)}) \geq (1-\delta) 2^{n(h(X,Y)-\varepsilon)}$.
4. Let $\tilde{X}$ and $\tilde{Y}$ be two independent random variables with the same marginal distributions of X and Y, i.e. $f_{\tilde{X}} = f_X$ and $f_{\tilde{Y}} = f_Y$. We have
$$Pr\{(\tilde{X}^n, \tilde{Y}^n) \in A_\varepsilon^{(n)}\} \simeq 2^{-nI(X;Y)}. \tag{6.22}$$
Formally, the following two bounds hold:
$$Pr\{(\tilde{X}^n, \tilde{Y}^n) \in A_\varepsilon^{(n)}\} \leq 2^{-n(I(X;Y)-3\varepsilon)}. \tag{6.23}$$
and $\forall \delta > 0$, n large,
$$Pr\{(\tilde{X}^n, \tilde{Y}^n) \in A_\varepsilon^{(n)}\} \geq (1-\delta) 2^{-n(I(X;Y)+3\varepsilon)}. \tag{6.24}$$

Proof. The proof is virtually identical to the proof of the AEP discrete theorem. □

We are now ready to state and prove the coding theorem for the AWGN channel, including both the direct and the converse part.

Theorem (*Capacity of the Gaussian channel)*.
A rate R is *achievable* if and only if

$$R < C = \max_{f_X(x): E[X^2] \leq P_X} I(X;Y), \tag{6.25}$$

and

$$C = \frac{1}{2}\log\left(1 + \frac{P_X}{P_N}\right) \qquad \text{bits/use of channel.} \tag{6.26}$$

Proof. The proof is organized in tree parts: in the first part we formally derive expression (6.26) for the Gaussian channel capacity, while in the second and in the third part we prove respectively the achievability and the converse parts of the theorem.

- Without loss of generality we can assume that the constraint in (6.25) holds at the equality. Then,

$$\begin{aligned} C = \max_{f_X(x):E[X^2]=P_X} I(X;Y) &= \max_{f_X(x):E[X^2]=P_X} h(Y) - h(Y|X) \\ &= \max_{f_X(x):E[X^2]=P_X} h(Y) - h(X+Z|X) \\ &\overset{(a)}{=} \max_{f_X(x):E[X^2]=P_X} h(Y) - h(Z|X) \\ &= \max_{f_X(x):E[X^2]=P_X} h(Y) - h(Z) \\ &= \max_{f_X(x):E[X^2]=P_X} h(Y) - \frac{1}{2}\log 2\pi e P_N, \end{aligned} \tag{6.27}$$

where in (a) we exploited the fact that $h(X+Z|X) = h(Z|X)$. We now look for a bound of $h(Y)$. For simplicity, we force Y to be a zero mean random variable (the entropy does not change); in this way, the variance of Y is [7]

$$\sigma_y^2 = E[Y^2] = E[X^2] + E[Z^2] \le P_X + P_N. \tag{6.28}$$

We know that, for a fixed variance, the Gaussian distribution yields the maximum value of the entropy. Hence, from (6.27),

$$\begin{aligned} h(Y) - \frac{1}{2}\log 2\pi e P_N &\le \frac{1}{2}\log 2\pi e(P_X + P_N) - \frac{1}{2}\log 2\pi e P_N \\ &= \frac{1}{2}\log\left[1 + \frac{P_X}{P_N}\right]. \end{aligned} \tag{6.29}$$

In order to conclude the proof we have to show an input distribution $f_X(x)$ exists which allows to reach the limit value. It's easy to see that such distribution is the Gaussian distribution with $\sigma_x^2 = P_X$. In this case, in fact, Y is also a Gaussian random variable, with $\sigma_y^2 = P_X + P_N$.

[7] Remember that the noise Z has zero mean.

- *(Achievability)*

We now pass to the proof of the direct implication of the theorem (stating that any rate below C is achievable). As usual, we make use of the concepts of random coding and joint typical decoding .
We consider the n-th extension of the channel. For a fixed rate R, the first step is the *generation of the codebook* for the 2^{nR} indexes. Since, as in the discrete case, we will consider large n values, we can generate the codewords i.i.d. according to a density function $f_X(x)$ with variance $P_X - \varepsilon$, so to ensure the fulfillment of the power constraint (according to the LGN, the signal power $(1/n)\sum_{i=1}^{n} x_i^2(1)$ tends to σ^2 as $n \to \infty$). Let $x^n(1), x^n(2), ..., x^n(2^{nR})$ be the codewords and $x^n(i)$ the generic codeword transmitted through the channel. The sequence y^n at the output of the channel is decoded at the receiver by using the same procedure described for the discrete channel decoding; that is, we search for a sequence which is jointly typical with the received one and we declare it to be the transmitted codeword.
We now evaluate the error probability. Without any loss of generality we assume that the codeword $W = 1$ was sent. Let us define the possible types of error:
- violation of the power constraint (tx side):

$$E_0 = \{x^n(1) : \frac{1}{n}\sum_{i=1}^{n} x_i^2(1) > P_X\}; \tag{6.30}$$

- the received sequence is not jointly typical with the transmitted one:

$$E_1 = \{(x^n(1), y^n) \notin A_\varepsilon^{(n)}\}; \tag{6.31}$$

- the received sequence is jointly typical with another sequence (different from the transmitted one):

$$E_i = \{(x^n(i), y^n) \in A_\varepsilon^{(n)}\}, \quad i \neq 1. \tag{6.32}$$

The error event, that is the event $\hat{W} \neq 1$, can be described as

$$\mathcal{E}_1 = E_0 \cup E_1^c \cup E_2 \cup ... \cup E_{2^{nR}}. \tag{6.33}$$

According to the code generation procedure used, the error probability averaged over all codewords and codes corresponds to the error probability for a transmitted codeword. Hence,

$$\begin{aligned} Pr(\mathcal{E}) &= Pr(\mathcal{E}|W=1) = Pr(\mathcal{E}_1) \\ &= P(E_0 \cup E_1^c \cup E_2 \cup ... \cup E_{2^{nR}}) \\ &\leq P(E_0) + P(E_1^c) + \sum_{1=2}^{2^{nR}} P(E_i). \end{aligned} \tag{6.34}$$

As $n \to \infty$, by the law of large number and the joint AEP theorem (respectively) we know that $P(E_0)$[8] and $P(E_1^c)$ tend to zero. Besides, we know that $X^n(1)$ and $X^n(i)$ for any $i \neq 1$ are independent by construction; then, the joint AEP theorem provides an upper bound to the probability that $X^n(i)$ and the output Y^n $(X^n(1) + Z^n)$ are jointly typical, which is $2^{-n(I(X;Y)-3\varepsilon)}$. Going on from (6.34), for sufficiently large n we have

$$\begin{aligned} Pr(\mathcal{E}) &\leq \varepsilon_1 + \varepsilon_2 + (2^{nR} - 2) \cdot 2^{-n(I(X;Y)-3\varepsilon)} \\ &\leq \varepsilon_1 + \varepsilon_2 + 2^{-n(I(X;Y)-R-3\varepsilon)}, \end{aligned} \tag{6.35}$$

with ε_1 and ε_2 arbitrarily small. If $R < I(X;Y)$, its easy to see that we can choose a positive ε such that $3\varepsilon < I(X;Y) - R$, thus yielding $2^{-n(I(X;Y)-R-3\varepsilon)} \to 0$ for $n \to \infty$ and then an arbitrarily small error probability. So far we have considered the average error probability; we can repeat the same passages of Section (5.2.3) in order to prove that the maximal probability of error λ_{max}^n is arbitrarily small too. Therefore, any rate below $I(X;Y)$ and then below C is achievable.

- *(Converse)*

We now show that the capacity of the channel C is the supremum of all achievable rates. The proof differs from the one given for the discrete case. For any code satisfying the power constraint we show that if $P_e^{(n)} \to 0$ then the rate R must be less then C.
Let W be a r.v. uniformly distributed over the index set $\mathcal{W} = \{1, 2, ..., 2^{nR}\}$. Being $H(W) = nR$ we can write

$$nR = I(W;Y^n) + H(W|Y^n). \tag{6.36}$$

[8] We point out that, strictly speaking, a new version of the AEP theorem, accounting also for the constraint on the power, is needed.

Applying Fano's inequality to $H(W|Y^n)$ yields [9]

$$\begin{aligned}
H(W|Y^n) &\leq 1 + P_e^{(n)} \log(|\mathcal{W}| - 1) \\
&< 1 + P_e^{(n)} \log(|\mathcal{W}|) \\
&= 1 + P_e^{(n)} nR \\
&= n\left(\frac{1}{n} + P_e^{(n)} R\right). \qquad (6.37)
\end{aligned}$$

Given that $P_e^{(n)} \to 0$ for $n \to \infty$, the expression in brackets can be made arbitrarily small for sufficiently large n. Let us name it ε_n. Then, we can write

$$\begin{aligned}
nR &\leq I(W;Y^n) + n\varepsilon_n \\
&\overset{(a)}{\leq} I(Y^n;X^n) + n\varepsilon_n \\
&= h(Y^n) - h(Y^n|X^n) + n\varepsilon_n \\
&\overset{(b)}{\leq} \sum_i h(Y_i) - h(Z^n) + n\varepsilon_n \\
&= \sum_i (h(Y_i) - h(Z_i)) + n\varepsilon_n, \qquad (6.38)
\end{aligned}$$

where (a) follows from the fact that $W \to x^n(W) \to Y^n$ is a Markov chain. Observing that $h(Y^n|X^n) = h(X^n + Z^n|X^n) = h(Z^n|X^n) = h(Z^n)$ (the signal and the noise are assumed independent) and that $h(Y^n) \leq \sum_i h(Y_i)$ (the received signals may not be independent), inequality (b) holds.

As usual, we can use the Gaussian density function as an upper bound for $h(Y_i)$, while Z_i is itself a Gaussian variable. We have: $E[Y_i^2] = E[X_i^2] + E[Z_i^2] + 2E[X_1]E[Z_i] = E[X_i^2] + E[Z_i^2]$, where $E[X_i^2]$ is the average power corresponding to the position i, i.e. $E[X_i^2] = \frac{1}{2^{nR}} \sum_w x_i(w)^2$ (The randomness of X_i directly follows from the randomness of W since $X_i = x_i(W)$). Let us denote it by P_i. Then: $E[Y_i^2] = P_i + P_N$ and from (6.38) we have

$$\begin{aligned}
&\leq \sum_i \left(\frac{1}{2}\log 2\pi e(P_i + P_N) - \frac{1}{2}\log 2\pi e P_N\right) + n\varepsilon_n \\
&\leq \sum_i \frac{1}{2}\log\left(1 + \frac{P_i}{P_N}\right) + n\varepsilon_n.
\end{aligned} \qquad (6.39)$$

[9] It can be proven that Fano's inequality also holds if the variable under investigation is discrete and the conditioning variable is continuous, but not in the reverse case.

Dividing by n we obtain:

$$R \;<\; \frac{1}{n}\sum_{i}^{n}\frac{1}{2}\log\left(1+\frac{P_i}{P_N}\right)+\varepsilon_n. \tag{6.40}$$

Since the log is a concave function we can exploit the following property:

Property. Let f be a concave function. The following relation holds

$$\frac{1}{n}\sum_{i=1}^{n} f(x_i) \leq f\left(\sum_{i=1}^{n}\frac{x_i}{n}\right). \tag{6.41}$$

Proof. The proof follows by induction. For $n = 2$ the relation is true, due to the concavity of f. Supposing that relation (6.41) is true for $n-1$, we have to prove that it also holds for n.
We can write:

$$\begin{aligned} f\left(\sum_{i=1}^{n}\frac{x_i}{n}\right) &= f\left(\frac{x_n}{n}+\frac{n-1}{n}\left(\frac{1}{n-1}\sum_{i=1}^{n-1}x_i\right)\right) \\ &\geq \frac{1}{n}f(x_n)+\frac{n-1}{n}f\left(\frac{1}{n-1}\sum_{i=1}^{n-1}x_i\right). \end{aligned} \tag{6.42}$$

Given the two points x_n and $\frac{1}{n-1}\sum_{i=1}^{n-1}x_i$, inequality (6.42) follows by the concavity of the function f. By applying relation (6.41) to the second term of the sum in (6.42) [10] we obtain

$$\begin{aligned} f\left(\sum_{i=1}^{n}\frac{x_i}{n}\right) &\geq \frac{1}{n}f(x_n)+\frac{n-1}{n}\frac{1}{n-1}\sum_{i=1}^{n-1}f(x_i), \\ &= \frac{1}{n}\sum_{i=1}^{n}f(x_i). \end{aligned} \tag{6.43}$$

□

By exploiting the property, from (6.40) we have

$$R < \frac{1}{2}\log\left(1+\frac{\sum_{i=1}^{n}P_i/n}{P_N}\right)+\varepsilon_n. \tag{6.44}$$

[10] Remember that we made the assumption that relation (6.41) holds for $n-1$.

Let us now observe that

$$\sum_{i}^{n} \frac{P_i}{n} = \frac{1}{n} \sum_{i}^{n} \frac{1}{2^{nR}} \sum_{w} x_i(w)^2 = \frac{1}{2^{nR}} \sum_{w} \left(\frac{1}{n} \sum_{i}^{n} x_i(w)^2 \right) < P_X, \quad (6.45)$$

where the expression in round brackets is the average power of the codeword $x^n(w)$ which averaged on all the codewords is less than P_X. We eventually get the following upper bound for the rate:

$$R < \frac{1}{2} \log \left(1 + \frac{P_X}{P_N} \right) + \varepsilon_n = C + \varepsilon_n. \quad (6.46)$$

This proves that for $n \to \infty$, if $P_e^{(n)} \to 0$, then necessarily $R \leq C$. □

6.4.3 Bandlimited Channels

Up to now we have treated the Gaussian channel as if it had an infinite bandwidth. In practice, this is never the case. Nevertheless, from Shannon'a sampling theorem we know that for a channel with finite bandwidth W, we can consider a sampled version of the signal with $T_c \leq 1/2W$ (T_c = sampling step). In this way the channel does not distort the transmitted signal.
If the *noise power spectral density* of the white noise is $N_0/2$ watts/hertz, we can say that $P_N = N_0 W$. Substituting it in the expression of the capacity we can say that

$$C = \frac{1}{2} \log \left(1 + \frac{P_X}{N_0 W} \right) \qquad \text{bits/transmission.} \quad (6.47)$$

By assuming a sampling frequency equal to the minimum frequency dictated by Shannon's sampling theorem, we can use the channel $2W$ times per second. Then, the channel capacity in bits per second is

$$C = W \log \left(1 + \frac{P_X}{N_0 W} \right) \qquad \text{bits/sec.} \quad (6.48)$$

where the ratio $P_X/N_0 W$ is the SNR (Signal to Noise ratio). This is the famous Shannon's formula for the *capacity of an additive white Gaussian noise channel (AWGN).*

Shannon's Capacity Curve

Looking at expression (6.48), the basic factors which determine the value of the channel capacity are the *channel bandwidth* W and the *input signal power* P_X. Increasing the input signal power obviously increases the channel capacity. However, the presence of the logarithm makes this growth slow. If we consider, instead, the channel bandwidth, which is the other parameter we can actually set, we realize that an increase of W (enlargement of the bandwidth) has two contrasting effects. On one side, a larger bandwidth allows to increase the transmission rate; on the other side, it causes a higher input noise at the receiver, thus reducing the capacity. While for small values of W it's easy to see that enlarging the bandwidth leads to an overall increase of the capacity, for large W we have[11]:

$$\lim_{W\to\infty} C = \log e \cdot \frac{P}{N_0}. \tag{6.49}$$

Then, by increasing only the bandwidth, we cannot increase the capacity beyond (6.49).
We now introduce Shannon's capacity curve which shows the existence of a tradeoff between power and bandwidth in any communication system.
Since in any practical reliable communication system we have $R < C$, the following relation is satisfied:

$$R < W \log\left(1 + \frac{P}{N_0 W}\right). \tag{6.50}$$

Dividing both sides by W yields

$$r < \log\left(1 + \frac{P}{N_0 W}\right), \tag{6.51}$$

where $r = R/W$ is the *spectral efficiency*, i.e. the number of bits per second that can be transmitted in a bandwidth unit (Hertz). By observing that $P = E_b \cdot R$ (we indicate by E_b the energy per transmitted bit) we get

$$r < \log\left(1 + r \cdot \frac{E_b}{N_0}\right). \tag{6.52}$$

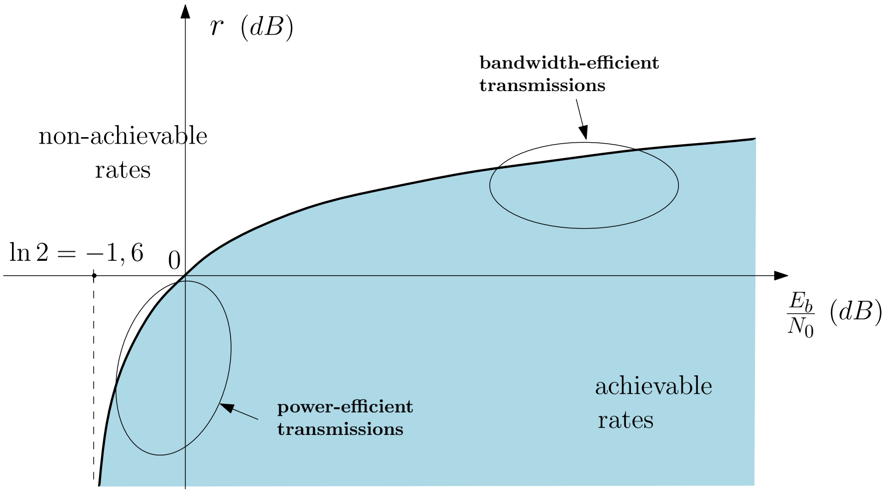


Figure 6.2: Shannon's capacity curve.

The above relation defines the achievable spectral efficiencies for any value of the ratio $\frac{E_b}{N_0}$. The locus of points r such that $r = \log\left(1 + r \cdot \frac{E_b}{N_0}\right)$ is the so called *Shannon's curve*. Shannon's curve splits the $(\frac{E_b}{N_0}, r)$ plane into two regions, as plotted in Figure 6.2. The region below the curve includes all the operative points for which reliable transmission is possible. In order to determine the behavior of the energy-to-noise ratio by varying r, we can raise both sides of equation (6.52) to the power 2, obtaining

$$\frac{E_b}{N_0} = \frac{2^r - 1}{r}. \tag{6.53}$$

Then, we can evaluate the following limit values:

- $r \to \infty \quad \Rightarrow \quad \frac{E_b}{N_0} \to \infty;$
- $r \to 0 \quad \Rightarrow \quad \frac{E_b}{N_0} \to \ln 2;$

proving that the curve in Figure 6.2 has a vertical asymptote in $\frac{E_b}{N_0} = \ln 2$, and below this value no reliable transmission is possible (for any value of r). Clearly, the more the working point is close to the curve, the more the communication system is efficient. All the communications whose main concern

[11] We make use of the approximation $\ln\left(1 + \frac{P}{N_0 W}\right) \approx \frac{P}{N_0 W}$ holding for $\frac{P}{N_0 W} << 1$.

is the limitation of the transmitted power lie on the area of the plane in which $r \ll 1$ (which is the area of the power efficient transmission). We refer to these system as *power-limited systems*. On the contrary, all the systems for which the bandwidth of the channel is small, referred to as *bandwidth-limited systems*, lie on the area where $r \gg 1$ (which is the area of the spectrally efficient transmission). Nevertheless, there is an unavoidable trade-off between *power efficiency* and *bandwidth efficiency*.
We now give some insights into how digital modulations are distributed with respect to Shannon's curve. From a theoretical point of view, the channel coding theorem asserts that it's possible to work with spectral efficiencies arbitrarily close to the curve with $P_e = 0$. In practice, classical modulation schemes have always a positive, although small, error probability and, despite this, they lie very far from the curve. Channel coding is what allows to improve the performance of a system, moving the operative points closer to Shannon's capacity curve.
Below, we see some examples of digital modulations. In the case of power-limited systems, high dimensional schemes are frequently used (e.g. M-FSK), which allow to save power at the expense of bandwidth. Contrarily, in the case of bandwidth-limited systems, the goal is to save bandwidth, then low dimensional modulation schemes (e.g. M-PSK) are often implemented[12].

- **B – PSK**

For a binary PSK (B-PSK or 2PSK), the error probability of a symbol corresponds to the bit error probability and is given by[13]

$$P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right). \tag{6.54}$$

For $P_e = 10^{-4}$ we get $E_b/N_0 = 8.5dB$ (from the table of the Q function). Let T_s indicate the transmission time of a symbol. By using a B-PSK modulation scheme we transmit one bit per symbol and then the per-symbol energy E_s corresponds to the energy per-bit E_b ($E_s = E_b$). Let W denote the bandwidth of the impulse of duration T_s, i.e. $W = 1/T_s$ [14]. Then $r = \frac{R}{W} = \frac{(1/T_s)}{1/T_s} = 1$.

[12] In all the examples we consider an error probability P_e of about 10^{-4}.

[13] The function Q gives the probability of the tail of the Gaussian distribution. More precisely, $Q(x)$ denote the the probability that a normal (Gaussian) random variable $\mathcal{N}(\mu, \sigma^2)$ will obtain a value larger than x standard deviations (σ) above the mean (μ).

[14] Strictly speaking, a finite impulse has an infinite bandwidth. Nevertheless, in digital modulation applications it is common to take the bandwidth as the frequency range which encompasses most of (but not all) the energy of the impulse. Indeed, the higher frequencies contribute at giving the (exact) shape, which, in such cases, is unnecessary.

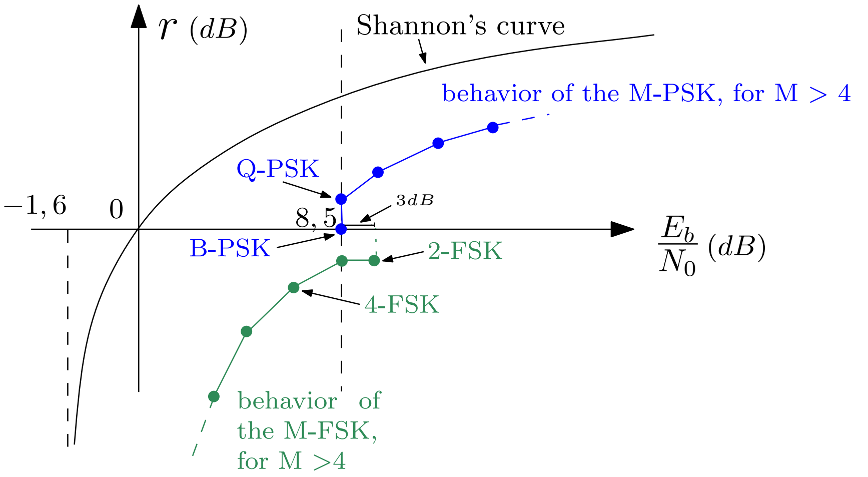


Figure 6.3: Location of the operative points of the classical modulation schemes on the Shannon's plane.

The corresponding operative point for the B-PSK scheme is shown in Figure 6.3.
According to Shannon's limit the same rate could have been reached with $E_b/N_0 = 0dB$ (with $8.5dB$ of power save).

- **Q – PSK**

In the QPSK modulation the probability of symbol error is approximated by

$$P_e \approx 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right), \tag{6.55}$$

where now $E_s = 2E_b$.
The multiplicative term in (6.55) does not influence the value of E_b/N_0 for $P_e = 10^{-4}$ which depends almost only on the argument of the Q function and then is the same as the BPSK case. On the other side, the QPSK modulation transmits two information bits per symbol ($R = 2/T_s$), hence $r = 2$ (see Figure 6.3).

- **M – PSK**

From the general expression for the P_e of a M-PSK it follows that as M grows we have an increase of E_b/N_0 (for the same value of P_e). Besides, the increase of the rate R with M is logarithmic ($\log M$ bits are transmitted simultaneously) and then the general expression for the spectral efficiency is

$r = \frac{(\log_2 M/T_s)}{1/T_s} = \log_2 M$. The approximate location of the operative points in the $(\frac{E_b}{N_0}, r)$ plane is illustrated in Figure 6.3.
As mentioned previously, phase modulations (low dimensionality modulations) permits to save bandwidth at the expense of power efficiency. Nevertheless, they remain far away from Shannon's curve.

- $\mathbf{M - FSK}$

Given a couple of orthogonal signals with energy E_s, the probability of error is given by $Q\left(\sqrt{\frac{E_s}{N_0}}\right)$. Considering M orthogonal signals, the union bound for the error probability yields

$$P_e \leq (M-1)Q\left(\sqrt{\frac{E_s}{N_0}}\right) = (M-1)Q\left(\sqrt{\log_2 M \frac{E_b}{N_0}}\right). \tag{6.56}$$

Neglecting the multiplicative term, for a modulation scheme with $M > 2$ we can save a factor $\log_2 M$ of bit energy E_b with respect the 2-FSK scheme (with the same P_e)[15]. However, orthogonal modulations are characterized by a linear growth of the bandwidth with M, i.e. $W = \frac{M}{T_s}$. Hence, since $R = \frac{\log_2 M}{T_s}$, we get $r = \frac{\log_2 M}{M}$. Saving power comes at the cost of a bandwidth increase. Figure 6.3 shows the operative points for various M.

The promise of Coding

Orthogonal modulation schemes approach the capacity curve as the dimensionality M grows but the price to pay is extremely high, since the bandwidth grows linearly with M. Starting from a simple example we show that through coding we can transmit information reliably, saving in power and without increasing too much the bandwidth.

Example.
Consider a situation in which we want to transmit 2 bits. Instead of using a $4 - PSK$ for transmitting the two bits in $2T_b$ sec, we can consider tree orthogonal signals in the interval $2T_b$, as depicted in Figure 6.4. The tree orthogonal signals, ψ_1, ψ_2 and ψ_3, constitute a basis for the three-dimensional space. We can then build four distinct waveforms to be associated to each

[15] Remember that for a 2-FSK $P_e = Q\left(\sqrt{\frac{E_b}{N_0}}\right)$, while for a 4-PSK $P_e = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$.

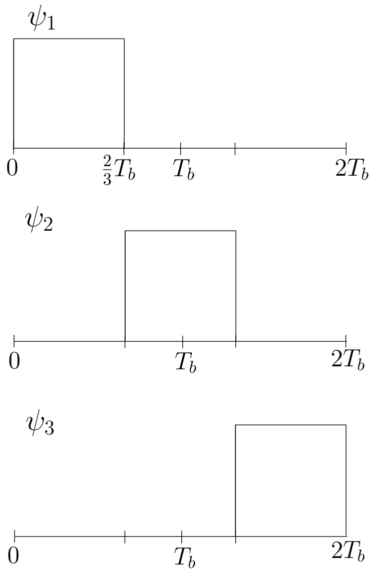


Figure 6.4: Tree orthogonal signals in $2T_b$.

of the starting configurations of two bits. For instance, the four waveforms could be the ones depicted in Figure 6.5. In vector notation:
$s_1 = \sqrt{E}(1, 1, 1)$;
$s_2 = \sqrt{E}(1, -1, -1)$;
$s_3 = \sqrt{E}(-1, 1, -1)$;
$s_4 = \sqrt{E}(-1, -1, 1)$,
where the signal energy is $E_s = 3E$[16]. The signal energy can be obtained from the bit energy as $E_s = 2E_b$ ($E = \frac{2}{3}E_b$).

We remind that the general approximation of the error probability as a function of the distance d among the transmitted signals is given by

$$P_e = Q\left(\sqrt{\frac{d^2}{2N_0}}\right). \tag{6.57}$$

Having increased the dimensionality of the system (from two to tree), the above procedure allows us to take four signals more distant from each other with respect to the Q-PSK scheme. In fact, taken an arbitrary couple of vectors in the constellation, we have

$$d^2 = 8E = \frac{16}{3}E_b > 4E_b, \tag{6.58}$$

[16] E indicate the energy of each pulse of duration $T = \frac{2}{3}T_b$ composing the signal.

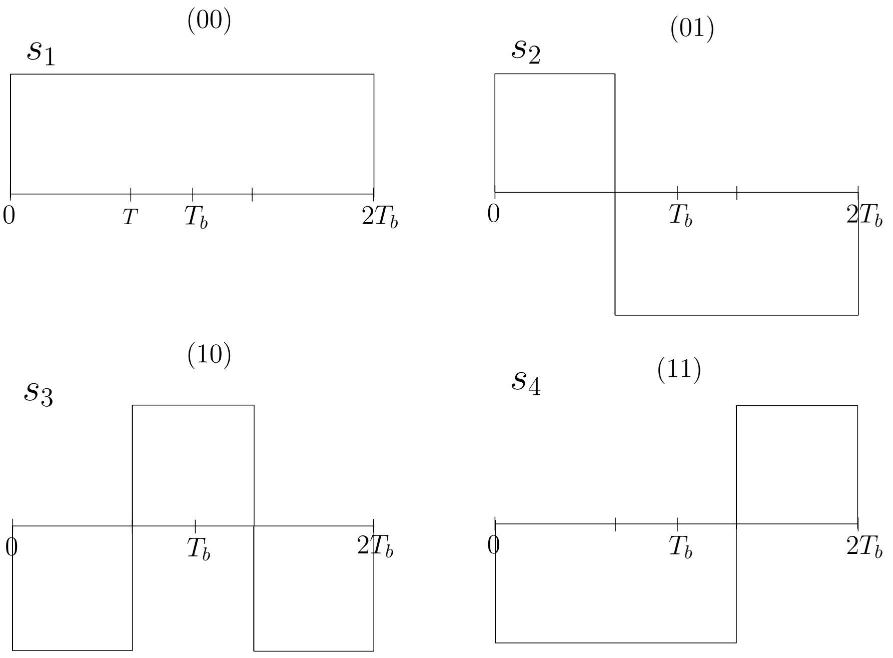


Figure 6.5: Possible waveforms we can associate to the configurations of two bits.

where $4E_b$ is the minimum distance between the signals in the Q-PSK constellation. Hence:

$$P_e = Q\left(\sqrt{\frac{8}{3}\frac{E_b}{N_0}}\right) = Q\left(\sqrt{\frac{4}{3}\frac{2E_b}{N_0}}\right), \tag{6.59}$$

leading to a coding gain of 4/3 with respect to the Q-PSK scheme. Nevertheless, the signals contain pulses narrower than T_b (whose pulse width is $T = \frac{2}{3}T_b$), and then they occupy larger bandwidth ($W = \frac{3}{2T_b}$). As a consequence, for this system we have $r = \frac{2}{3}$ [17]. Therefore, there is always a trade-off between power and bandwidth but in this case the trade-off is more advantageous, as the following generalization of the above procedure clarifies.

Generalized procedure

What we have described above is nothing but a primitive form of coding. Let us now suppose that we aim at transmitting k bits. We can use a code $C(k, n)$ in order to associate to any configuration of k bits another configuration of n bits (with $n > k$). In this way the constellation of k points can be represented in the n-dimensional space where each point lies

[17] With a Q-PSK we would have $r = 1$.

on a vertex of a n-dimensional hypercube. The vector representation of a given point is of the following kind: $s = \sqrt{E}(..., 1, .., -1, ..)$. Since the error probability is dominated by the contribution given by the couple of points at the shortest distance, we consider again the following approximation:

$$P_e \simeq Q\left(\sqrt{\frac{d_{min}^2}{2N_0}}\right). \tag{6.60}$$

According to this procedure, we have $E = \frac{k}{n}E_b$. Indicating with d_H the Hamming distance between two codewords (number of positions in which two codewords differ), it's straightforward to deduce that the distance between two codewords has the expression $d^2 = 4d_H E = 4d_H \frac{k}{n} E_b$. Then, denoting by d_{Hmin} the minimum Hamming distance between two codewords we can write

$$P_e \simeq Q\left(\sqrt{\frac{2E_b}{N_0}\left(d_{Hmin} \cdot \frac{k}{n}\right)}\right), \tag{6.61}$$

where $G_c = d_{Hmin} \cdot \frac{k}{n}$ is the so called *coding gain*.
In turn, the bandwidth becomes $W = \frac{n}{kT_b}$ and then $r = \frac{k}{n}$. Therefore, a high redundancy $(n - k)$ causes a significant bandwidth expansion.
It is evident that for a given ratio k/n the best approach to coding is to generate the codewords with the *largest minimum distance* (d_{Hmin}), since this parameter does not affect the bandwidth. Nevertheless, if we want a larger d_{Hmin} in order to increase the coding gain G_c, we need to add redundancy and then decrease the ratio k/n.
Through coding, as through FSK modulation schemes, we save power at the expense of bandwidth; but now the *exchange rate* is much more advantageous, as Figure 6.6 shows.
Since Shannon's time, great efforts have been made by researches for designing channel coding schemes that get as close as possible to Shannon's limit. The recent invention of the LDPC and Turbo codes finally allowed to get very close to reach this goal.

Ex: Prove that by using a repetition code we have a $G_c = 1$, while for the Hamming code $G_c = 3$.

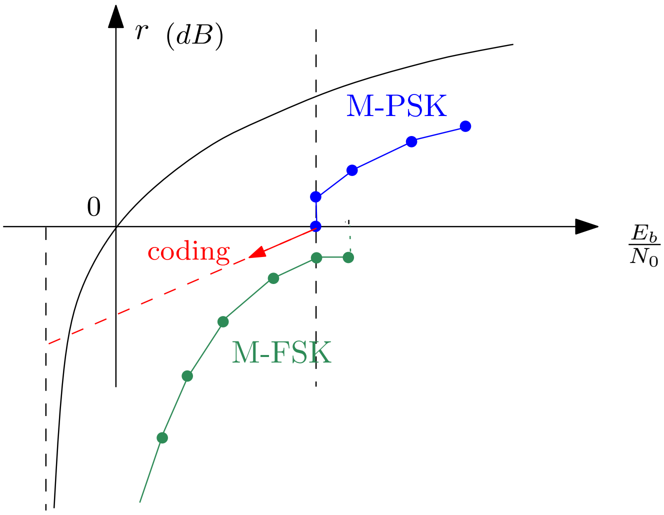


Figure 6.6: The role of coding in the $(\frac{E_b}{N_0}, r)$ plane.

Chapter 7

Rate Distortion Theory

The source coding theorem states that a discrete source X can be encoded lossless as long as $R \geq H(X)$. However, in many real applications, the presence of (a moderate amount of) reconstruction errors does not compromise the result of the transmission (or the storage); then, sometimes, it may be preferable to admit errors within certain limits, i.e. a quality loss, to increase compression efficiency. In other words, lossless compression removes the statistical redundancy, but there are other types of redundancy, e.g. psychovisual and psychoacoustic redundancy (depending on the application), that can be taken into account in order to increase the compression ratio. Think for instance to JPEG compression for still images!

In order to introduce a controlled reduction of quality we need to define of a *distortion measure*, that is a measure of the distance between the random variable and its (lossy) representation. The basic problem tackled with by rate distortion theory is determining the minimum expected distortion which is necessary to tolerate in order to compress the source at a given rate.

Rate distortion theory is particularly suited to deal with continuous sources. We know that in the continuous case lossless coding cannot be used because of the fact that a continuous source requires an infinite precision to be represented exactly. Then, while for discrete sources the rate distortion theory can be introduced as an additional (optional) tool to source coding, for continuous sources it is an essential tool for representing the source.

7.1 Rate Distortion Function

Let us consider, without loss of generality, a source X with finite alphabet $\mathcal{X}$, $X \sim p_X(x)$[1]. Let $\hat{X}$ denote the (lossy) reconstruction of the random

[1] Similar arguments hold for continuous sources.

variable X, with finite alphabet $\hat{\mathcal{X}}$.
We introduce the distance measure $d(x,\hat{x})$; the most commonly used for continuous and discrete alphabets are respectively the Euclidean and the Hamming distance:

- Euclidean distance:
$$d(x,\hat{x}) = (x-\hat{x})^2; \tag{7.1}$$
- Hamming distance:
$$d(x,\hat{x}) = x \oplus \hat{x} = \begin{cases} 0 & \text{se } x = \hat{x} \\ 1 & \text{se } x \neq \hat{x}. \end{cases} \tag{7.2}$$

The *distortion measure* or *distortion function* D is defined as

$$D = E\left[d(X,\hat{X})\right], \tag{7.3}$$

where the average is taken over all the alphabet symbols x and all the possible values of the reconstruction $\hat{x}$. In the Euclidean case the distortion function is

$$D = E\left[(X-\hat{X})^2\right], \tag{7.4}$$

i.e. the *mean square error* between the signal and its reconstruction.
In the Hamming case the distortion function is

$$D = E\left[X \oplus \hat{X}\right] = P_e, \tag{7.5}$$

i.e. the *probability of a reconstruction error.*
We can extend the definition of distance to sequences of symbols x^n and $\hat{x}^n$ as follows:

$$d(x^n,\hat{x}^n) = \frac{1}{n}\sum_{i=1}^{n} d(x_i,\hat{x}_i), \tag{7.6}$$

which, for stationary sources, leads to the same mean value as before, that is

$$D = E\left[d(X^n,\hat{X}^n)\right] = E\left[d(X,\hat{X})\right]. \tag{7.7}$$

Having introduced the above quantities, we can give the following definition.

Definition. The *rate distortion function* $R(D)$ gives the minimum number of bits (R_{min}) guaranteeing a reconstruction distortion $E\left[d(X,\hat{X})\right] \leq D$.

Note: for $D = 0$, no distortion is accepted, hence $R(0)$ is exactly the entropy of the source $H(X)$[2].

It's easy to argue that $R(D)$ is a monotonic decreasing function of D. Then, we can also compute the inverse function $D(R)$, named the *distortion rate function*. Given the (maximum) number of bits we are willing to spend, $D(R)$ tells us the minimum amount of distortion which is introduced in the reconstruction.

The main theorem of the rate distortion theory (Shannon, 1959), also known as the *lossy coding theorem*, is the following.

Theorem (*Rate Distortion*).
Let $X \sim p(x)$. Then

$$R(D) = \min_{p(\hat{x}|x):E[d(X,\hat{X})] \leq D} I(X;\hat{X}) \tag{7.8}$$

is the minimum achievable rate at distortion D.

Observe that the conditional distribution $p(\hat{x}|x)$ in (7.8) is the actual degree of freedom in the minimization; it derives from the joint distribution $p(\hat{x}, x)$ by exploiting the knowledge of $p(x)$ (conditional probability theorem). Even though for simplicity we refer to discrete sources, we stress that the theorem holds both for discrete and continuous sources.
The rigorous proof of the theorem is beyond the scope of these notes. Instead, we'll provide an outline of the proof in order to point out the main ideas behind it.
Before starting, it's necessary to extend the typicality definitions given in Chapter 4 so to take into account the distortion D. In the new context addressed here, in fact, we aim to characterize sequences which are typical even with respect to a given distortion measure, namely 'distortion typical sequences'.

Definition. Let X be a discrete memoryless source with pmf $p(x)$ and let $\hat{X}$ be the reconstructed source with pmf $p(\hat{x})$[3]. Let $p(x, \hat{x})$ be the joint probability distribution.

[2] By referring to continuous sources, $R(0)$ is ∞.

[3] For notational simplicity we omit the subscript in $p_X(x)$ and $p_{\hat{X}}(\hat{x})$, being it recoverable from the argument.

We define the *distortion jointly typical set* $A_{d,\varepsilon}^{(n)}$ as follows

$$
\begin{aligned}
A_{d,\varepsilon}^{(n)} = \Big\{ &(x^n, \hat{x}^n) \in \mathcal{X}^n \times \hat{\mathcal{X}}^n : \\
&\left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \varepsilon, \left| -\frac{1}{n} \log p(\hat{x}^n) - H(\hat{X}) \right| < \varepsilon, \\
&\left| -\frac{1}{n} \log p(x^n, \hat{x}^n) - H(X, \hat{X}) \right| < \varepsilon, \\
&\Big\| d(x^n, \hat{x}^n) - E[d(X, \hat{X})] \Big\| < \varepsilon \Big\}, \qquad (7.9)
\end{aligned}
$$

representing, respectively, the typicality of x^n and $\hat{x}^n$ with respect to $p(x)$ and $p(\hat{x})$, the joint typicality and the typicality with respect to distortion.

Note that the difference from the previous definition of jointly typical set resides only in the additional constraint which expresses the typicality of the couples of sequences with respect to distortion. Instead of a probability, the involved statistics for measuring this type of typicality is the distance between the random variables. Let us define $d(x_i, \hat{x}_i) = d_i$ and consider the corresponding random variable D_i, which is a function of the random variables X_i and $\hat{X}_i$, i.e. $D_i = d(X_i, \hat{X}_i)$. By applying the law of large numbers, as $n \to \infty$ the sample mean of d_i tends to the ensemble average, that is

$$\frac{1}{n} \sum_{i=1}^{n} D_i \underset{n\to\infty}{\longrightarrow} E[D] \quad \text{in prob.} \qquad (7.10)$$

Then, the additional requirement regarding distortion does not limit much the number of sequences in $A_{d,\varepsilon}^{(n)}$ with respect to the number of sequences in the jointly typical set, since for large n a sequence belongs to $A_{d,\varepsilon}^{(n)}$ with probability arbitrarily close to 1.

Outline of the Proof.

- *(Direct implication/Achievability)*

To start with, let us fix $p(\hat{x}|x)$[4]. Knowing the marginal pmf $p(x)$, from $p(\hat{x}|x)$ we can derive the joint pmf $p(\hat{x}, x)$ and then $p(\hat{x})$.
Fix also R.
The proof of achievability proceeds along the following steps.

[4] Chosen according to the constraint $E[d(X, \hat{X})] \leq D$.

Generation of the codebook. Generate a codebook $\mathcal{C}$ consisting of 2^{nR} sequences $\hat{x}^n$ drawn i.i.d. according to $p(\hat{x})$.
Encoding. For any x^n find a sequence $\hat{x}^n$ such that $(x^n, \hat{x}^n) \in A^{(n)}_{d,\varepsilon}$. If there is more than one such x^n, take the one with the least index. If there is no such sequence take the first sequence, i.e. $\hat{x}^n(1)$, and declare an error. The index $i = \{1, 2, ..., 2^{nR}\}$ of the chosen sequence is the codeword.
Decoding. The list of the sequences $\hat{x}^n$ is known to the decoder. Then, the decoder associates to the received index i the corresponding sequence $\hat{x}^n(i)$.
Computation of the distortion. Consider the expected distortion over the random choice of codebooks, i.e. $E_{X^n,\mathcal{C}}[d(X^n, \hat{X}^n)]$ (the subscript of E indicates the variables over which the expectation is taken).

By referring to the above procedure it is possible to prove that:

$$E_{X^n,\mathcal{C}}[d(X^n, \hat{X}^n)] \leq D \quad \text{if} \quad R \geq I(X, \hat{X})^5. \tag{7.11}$$

We now give an intuitive view of the implication in (7.11). From the initial choice of $p(\hat{x}, x)$ and from the definition of $A^{(n)}_{d,\varepsilon}$ we argue that if the coder at the transmitter side has found a distortion jointly typical sequence, then the expected distortion is close to D. But the sequences $\hat{x^n}$ are drawn only accordingly to the marginal distribution $p(\hat{x})$ and not to the joint one. Therefore, we have to evaluate the probability that a pair of sequences $(x^n, \hat{x}^n)$ (generated by the correspondent marginal distributions) is typical. According to the joint AEP theorem $Pr\{((x^n, \hat{x}^n))\} \sim 2^{-nI(X;\hat{X})}$. Hence, the probability of finding at least one $\hat{x}^n$ which is distortion typical with x^n during the encoding procedure is

$$2^{nR} \cdot 2^{-nI(X;\hat{X})} = 2^{-n(I(X;\hat{X})-R)}.$$

We can hope of finding such sequence only if $R > I(X, \hat{X})$. If this is not the case, the probability of finding a typical x^n tends to zero as $n \to \infty$.
Now, we can exploit the degree of freedom we have on $p(\hat{x}|x)$ in order to determine the minimum rate at which reconstruction is possible along with the fixed maximum distortion D. Hence: $R_{min} = R(D) = \min_{p(\hat{x}|x):E[d(x,\hat{x})]\leq D} I(X; \hat{X})$.

- *(Reverse implication/Converse)*

The proof is quite involved, so we do not sketch it in these notes.

[5] Note the correspondence with the channel capacity proof in which we show that $P_e \to 0$ if $R < I(X, Y)$.

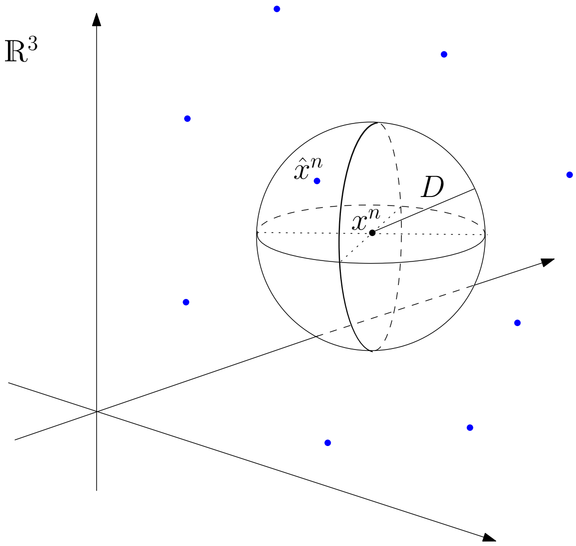


Figure 7.1: Graphical representation of the lossy coding procedure (quantization) for $n = 3$. The choice $n = 3$ is only to ease the graphical representation.

□

Note: in the rate distortion theorem we have considered the average distortion $E[d(X, \hat{X})]$. Nevertheless, the same result holds by considering a stricter distortion constraint.

7.1.1 Interpretation of the Rate Distortion Theorem

The rate distortion theorem states that the function $R(D)$ in (7.8) specifies the lowest rate at which the output of a source can be encoded while keeping the distortion less than or equals to D. The sketch of the proof allows us to make some interesting considerations. The generation of the codebook is nothing else than 'quantizing' blocks of source symbols. Indeed, the proof considers a finite set of 2^{nR} values, $\{\hat{x}^n\}$, to represent the sequences of symbols x^n. The concept arises naturally if we consider a continuous source X ($x^n \in \mathbb{R}^n$). Let R be the number of bits used to represent a symbol of the source, then nR bits are associated to a n-length sequence of symbols. Figure 7.1 illustrates the quantization procedure. According to the distortion constraint, for any sequence x^n the search of the point $\hat{x}^n$ is restricted to the neighborhood of x^n for which $d(x^n, \hat{x}^n) \leq D$. If the quantization is sufficiently dense, that is if $R > I(X, \hat{X})$, with high probability it's possible to

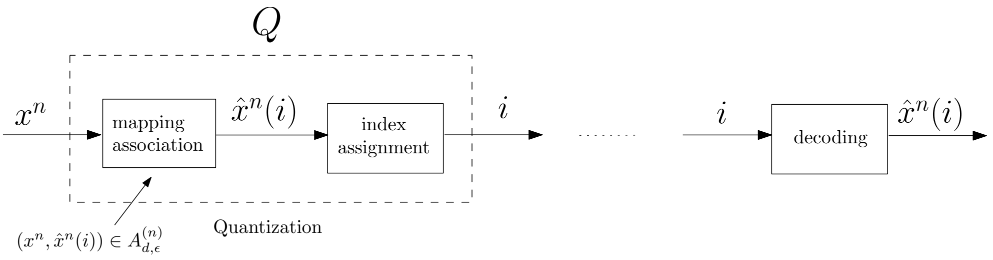


Figure 7.2: Lossy source coding: the Shannon scheme.

find a point $\hat{x}^n$ in the neighborhood of x^n, which satisfies the jointly typicality property for large n. The quantization can be more or less coarse depending on the value of D. If the tolerable distortion is small the quantization must be fine (R large), while it can be made coarser (smaller R) as the amount of tolerable distortion increases. Indeed, looking at the figure, if D decrease we would have to increase the number of reconstruction sequences $\hat{x}^n$ in order that at least one of them ($\hat{x}^n$) falls inside each region with high probability. Figure 7.2 schematically represents the lossy coding procedure.

It's easy to argue that in the discrete source case ($x^n \in \mathcal{X}^n$) the same procedure leads to a further quantization of an already quantized signal, but nothing conceptually changes. We stress again that in this case, as opposed to the continuous source case, lossless coding is possible. However, rate distortion theory can be applied whenever we prefer to decrease the coding rate at the price of an introduction of an acceptable distortion.
As it happened for the proofs of Source and Channel Coding Theorems, the proof of the Rate Distortion Theorem does not indicate a practical coding strategy. Therefore, we have to face with the problem of finding the optimum set of points $\{x^n\}$ to represent the source for finite n. To this purpose, it's easy to guess that knowing the type of source helps and then should be taken into account in order to make the rate close to the theoretical value $R(D)$. This problem will be faced with in Section 7.2.

7.1.2 Computing the Rate Distortion Function

We now compute the rate distortion function $R(D)$ for some common sources.

Bernoulli source

$$\begin{aligned} p(1) &= p; \\ p(0) &= 1 - p. \end{aligned}$$

Let us suppose, without any loss of generality, that $p \leq 1/2$.
The most natural choice for the distance measure is the Hamming distance d_H.
The rate distortion function for the Bernoulli source with parameter p is given by

$$R(D) = \begin{cases} h(P) - h(D) & \text{if } D \leq p^6 \\ 0 & \text{if } D > p. \end{cases} \tag{7.12}$$

Proof. We have to compute:

$$\min_{p(\hat{x}|x): E[X \oplus \hat{X}] \leq D} I(X; \hat{X}). \tag{7.13}$$

Case 1: $D > p$

Let us take $\hat{X} = 0$. This choice allows to achieve the lower bound for the mutual information, i.e. $I(X, \hat{X}) = 0$. We have to check if the constraint is satisfied. It is easy to argue that it is so since $E[X \oplus 0] = p(x = 1) = p < D$. Note that this solution is also suggested by intuition: a reconstruction with an error less than or equal to a value (D) greater than p is trivially obtained by encoding every sequence as a zero sequence.

Case 2: $D \leq p$

It is possible to solve minimization (7.13) through the same procedure adopted for the computation of the capacity for some simple channels in Section 5.2.4: first we find a lower bound for $I(X; \hat{X})$, later on we seek a distribution which fulfils the constraint and attains the limit value.

[6] Notice that the notation $h(D)$ is correct since $D = E[d_H] \leq 1$.

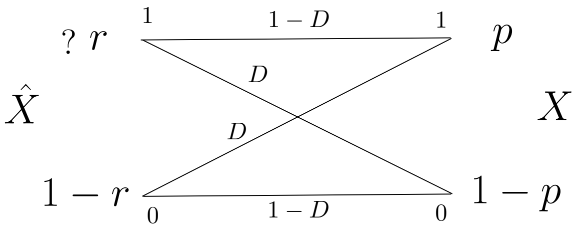


Figure 7.3: Joint distribution between $\hat{X}$ and X given by the binary symmetric channel (test channel).

Let us find the lower bound:

$$\begin{aligned} I(X;\hat{X}) &= H(X) - H(X|\hat{X}) \\ &\overset{(a)}{=} h(p) - H(X \oplus \hat{X}|\hat{X}) \\ &\geq h(p) - H(X \oplus \hat{X}) \\ &\overset{(b)}{=} h(p) - h(E[X \oplus \hat{X}]), \end{aligned} \tag{7.14}$$

where (a) follows from the fact that $x = \hat{x} \oplus (x \oplus \hat{x})$, while (b) is obtained by observing that $X \oplus \hat{X}$ is itself a binary source with $Pr\{X \oplus \hat{X} = 1\} = E[(X \oplus \hat{X})]$.
Now, since the binary entropy $h(r)$ grows with r (with $r < 1/2$) and $E[(X \oplus \hat{X})]$ is less than D, we have $h(E[X \oplus \hat{X}]) \leq h(D)$ and then by going on from (7.14) we get

$$I(X;\hat{X}) \geq h(p) - h(D). \tag{7.15}$$

At this point we know that $R(D) \geq h(P) - h(D)$. Let us show that a conditional probability distribution $p(\hat{x}|x)$ attaining this value exists.
For establishing a relation between the two binary random variables X and $\hat{X}$, that is for determining a joint distribution, we can refer to the binary symmetric channel (BSC) with $\varepsilon = D$, see Figure 7.3. Let us determine the input of the channel $\hat{X}$ so that the output X is the given distribution (with the fixed $p(x)$). Let $r = Pr(\hat{X} = 1)$. Then, we require that

$$r(1 - D) + (1 - r)D = p, \tag{7.16}$$

that is

$$r = \frac{p - D}{1 - 2D}. \tag{7.17}$$

For $D \leq p$ the choice $p(\hat{x} = 1) = \frac{p-D}{1-2D}$:

1. satisfies the constraint, being $E[X \oplus \hat{X}] = Pr\{X \neq \hat{X}\} = D$;
2. reaches the minimum value $I(X;Y) = h(p) - h(D)$ by construction.
We have then proved that $R(D) = h(p) - h(D)$. □

Gaussian source

Let X be a Gaussian source, $X \sim \mathcal{N}(0, \sigma_X^2)$.
For this type of source it is reasonable to adopt the Euclidean distance (squared error distortion). The rate distortion function is given by

$$R(D) = \begin{cases} \frac{1}{2}\log\frac{\sigma_x^2}{D} & \text{if } D \leq \sigma_x^2 \\ 0 & \text{if } D > \sigma_x^2. \end{cases} \tag{7.18}$$

Proof. We have to compute:

$$\min_{f(\hat{x}|x): E[(X-\hat{X})^2] \leq D} I(X;\hat{X}). \tag{7.19}$$

<u>Case 1</u>: $D > \sigma_x^2$

We can take $\hat{X} \equiv 0$. With this choice the average error we make is the variance of the random variable X which, being less then D, permits to satisfy the constraint: $E[X^2] = \sigma_x^2 \leq D$ [7]. Besides, this choice attains the absolute minimum for $I(X;\hat{X})$, that is $I(X;\hat{X}) = 0$. Then, $R(D) = 0$.

<u>Case 2</u>: $D \leq \sigma_x^2$

We go along the same steps we followed in the Bernoulli case.
We first find a lower bound for $I(X;\hat{X})$:

$$\begin{aligned} I(X;\hat{X}) &= h(X) - h(X|\hat{X}) \\ &= \frac{1}{2}\log 2\pi e\sigma_x^2 - h(X-\hat{X}|\hat{X}) \\ &\geq \log 2\pi e\sigma_x^2 - h(X-\hat{X}) \\ &\overset{(a)}{\geq} \log 2\pi e\sigma_x^2 - \log 2\pi e E[(X-\hat{X})^2], \\ &\geq \frac{1}{2}\log\frac{\sigma_x^2}{D}, \end{aligned} \tag{7.20}$$

where (a) derives from the fact that $(X - \hat{X})$ is a random variable whose

[7] We remind that for any random variable Z the relation $\sigma_z^2 = E[Z^2] - \mu_z^2$ holds.

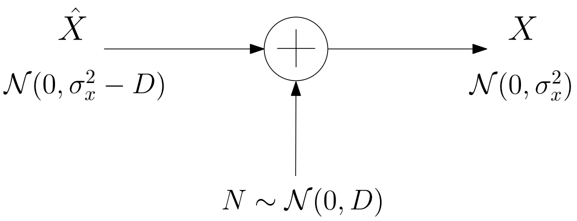

$\hat{X}$
$\mathcal{N}(0, \sigma_x^2 - D)$
X
$\mathcal{N}(0, \sigma_x^2)$
$N \sim \mathcal{N}(0, D)$

Figure 7.4: Joint distribution between $\hat{X}$ and X given by the AWGN (test channel).

variance is surely less than the mean square error $E[(X-\hat{X})^2]$ and then, the entropy of a Gaussian random variable with variance $E[(X-\hat{X})^2]$ gives an upper bound for $h(X-\hat{X})$ (principle of the maximum entropy).

We now have to find a distribution $f(\hat{x}|x)$ that attains the lower bound for I. As before, it is easier to look at the reverse conditional probability $f(x|\hat{x})$ as the transitional probability of a channel and choose it in such a way that the distribution of the channel output x is the desired one. Then, from the knowledge of $f(x)$ and $f(\hat{x})$ we derive $f(\hat{x}|x)$. Let us consider the relation between $\hat{X}$ and X depicted in Figure 7.4 (*test channel*), i.e. we assume that the difference between X and its reconstruction $\hat{X}$ is an additive Gaussian noise N. It is easy to check that this choice:

1. satisfies the distortion constraint; indeed

$$E[(X-\hat{X})^2] = E[N^2] = D \tag{7.21}$$

2. achieves the lower bound:

$$\begin{aligned} I(X,\hat{X}) &= h(X) - h(X|\hat{X}) \\ &= \frac{1}{2}\log 2\pi e\sigma_x^2 - h(X-\hat{X}|\hat{X}) \\ &= \frac{1}{2}\log 2\pi e\sigma_x^2 - h(N|\hat{X}) \\ &\overset{(a)}{=} \frac{1}{2}\log 2\pi e\sigma_x^2 - h(N) \\ &= \frac{1}{2}\log 2\pi e\sigma_x^2 - \frac{1}{2}\log 2\pi e D \\ &= \frac{1}{2}\log \frac{\sigma_x^2}{D}, \end{aligned} \tag{7.22}$$

where (a) follows from the fact that according to the chosen model N and $\hat{X}$ are independent. □

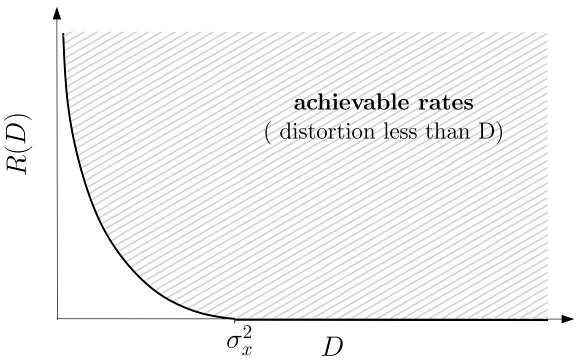


Figure 7.5: Rate distortion curve for a Gaussian source.

Figure 7.5 depicts the rate distortion curve for a Gaussian source. The curve partitions the space into two regions; by varying D, only the rates lying above the curve are achievable. For $D \to 0$ we fall back into lossless source coding, and then $R \to \infty$ (entropy of a continuous random variable). If instead the reconstruction distortion is larger than σ_x, there is no need to transmit any bit ($R = 0$).
For the Gaussian source we can express the distortion in terms of the rate by reversing $R(D)$, obtaining

$$D(R) = \sigma_x^2 2^{-2R}. \tag{7.23}$$

Given the number of bits we are willing to spend for describing the source, $D(R)$ provides the minimum distortion we must tolerate in the reconstruction (Figure 7.6). Obviously, the condition $D = 0$ is achievable only asymptotically.
Let us evaluate the signal to noise ratio associated to the rate distortion:

$$SNR = \frac{\sigma_x^2}{D} = 2^{2R} \to SNR_{db} = 6R. \tag{7.24}$$

For any bit we add, the SNR increases by 6 dB.

<u>Note</u>: it is possible to prove that, like the differential entropy for the Gaussian source, the rate distortion function for the Gaussian source is larger than the rate distortion function for any other continuous source with the same variance. This means that, for a fixed D, the Gaussian source gives the maximum $R(D)$. This is a valuable result because for many sources the computation of the rate distortion function is very difficult. In these cases,

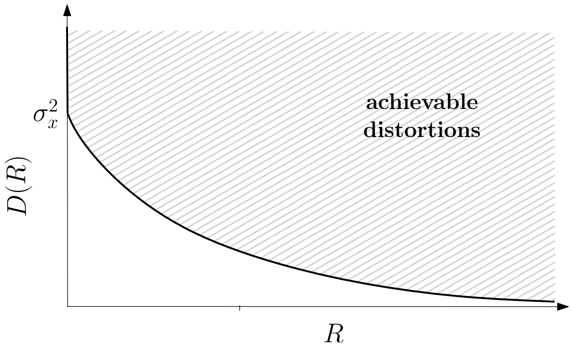


Figure 7.6: Distortion rate curve for a Gaussian source. Fixed R, the amount of distortion introduced cannot be less than the value of the curve in that point.

the rate distortion curve in Figure 7.5 provides an upper bound.

7.1.3 Simultaneous representation of Independent Gaussian Random variables

Let us consider the problem of representing M independent zero mean Gaussian random variables $X_1, ..., X_M$ having different variances, i.e. $X_i \in \mathcal{N}(0, \sigma_i^2)$, given a global distortion constraint:

$$E\left[\sum_{i=1}^{M} d_i(X_i, \hat{X}_i)\right] \leq D. \tag{7.25}$$

We take $d_i(X_i, \hat{X}_i) = (X - \hat{X})^2$ (squared-error distortion).
The problem we have to solve is the following: given R bits for representing the M sources, what is the best possible allocation, that is the allocation which minimizes the overall distortion D?
Because of the global constraint we have to join the M random variables in a vector and encode it as a unique symbol. We then have to consider the extension of the rate distortion function to the vector case, that is:

$$R(D) = \min_{f(\hat{x}^M|x^M):E\left[||X^M - \hat{X}^M||^2\right] \leq D} I(X^M, \hat{X}^M), \tag{7.26}$$

where we used the Euclidean norm.
As usual, firstly we determine the lower bound and later search for a joint distribution which reaches it.

1. Evaluation of the lower bound:

$$\begin{aligned}
I(X^M;\hat{X}^M) =& h(X^M) - h(X^M|\hat{X}^M) \\
=& \sum_{i}^{M} h(X_i) - \sum_{i=1}^{M} h(X_i|\hat{X}^M, X_{i-1}, ..., X_1) \qquad (7.27) \\
\geq& \sum_{i}^{M} h(X_i) - \sum_{i=1}^{M} h(X_i|\hat{X}_i) \qquad (7.28) \\
=& \sum_{i=1}^{M} I(X_i, \hat{X}_i) \\
\geq& \sum_{i=1}^{M} R(D_i) \qquad (7.29) \\
=& \sum_{i=1}^{M} \left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+, \qquad (7.30)
\end{aligned}$$

where in equality (7.27) we exploited the independence of the random variables X_i (for the first term) and the chain rule (for the second term), while inequality (7.28) follows from the fact that conditioning reduces entropy. In equality (7.30), the plus sign in the subscript is a compact way for writing the expression in (7.18) (each term of the sum is the expression in round brackets if it is positive, 0 otherwise).
Each term D_i, $i = 1, ..., M$ denotes the average distortion assigned to the i-th variable ($D_i = E[(X_i - \hat{X}_i)^2]$). Overall we must have $\sum_{i=1}^{M} D_i \leq D$.

2. Search for a conditional probability $f(x^M|\hat{x}^M)$: to do so we wonder when inequalities (7.28) and (7.29) are satisfied to equality. As to the former, since X_i does not depend on $X_i, ..., X_{i-1}$ we have that $h(X_i|\hat{X}^M, X_i, ..., X_{i-1}) = h(X_i|\hat{X}^M)$. Hence, by choosing $f(x^M|\hat{x}^M) = \prod_{i=1}^{M} f(x_i|\hat{x}_i)$[8] equation (7.28) holds at equality. We still have the freedom of choosing the probabilities $f(x_i|\hat{x}_i)$ for each i; we can take them in such a way that equation (7.29) holds at equality too. From the previous evaluation of $R(D)$ for the Gaussian source, we know that if we consider the conditional probability $f(x_i|\hat{x}_i)$ obtained by the test channel which adds noise $N \sim \mathcal{N}(0, D_i)$ to an input $\hat{x}_i \sim \mathcal{N}(0, \sigma_i^2 - D)$, we achieve the condition $I(X_i; \hat{X}_i) = R(D_i)$. Hence, taking $f(x_i|\hat{x}_i)$ for each i in this way permits to satisfy (7.28) at the equality.

[8] According to this expression, given the reconstruction $\hat{x}_i$ the symbol x_i is conditionally independent on the other reconstructions.

We have then found a $f(x^M|\hat{x}^M)$ such that $I(X^M;\hat{X}^i) = \sum_{i=1}^{M}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+$. Now we remember that in our problem the distortion values D_i, $i = 1, ..., M$ provide an additional degree of freedom we can exploit. Hence, from (7.30) the final minimum is obtained by varying D_i, $i = 1, ..., M$, that is:

$$R(D) = \min_{D_i:\sum_i D_i = D} \sum_{i=1}^{M}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+ . \tag{7.31}$$

In (7.31) the distortion constraint is expressed with equality since it is reasonable to expect that the minimum value will be achieved exploiting all the available distortion.
Then, in order to find the rate distortion function for the M independent Gaussian random variables with global distortion constraint we have to solve a constrained optimization problem.

We can solve the minimization in (7.31) by applying the Lagrange method. Accordingly, we have to minimize the functional

$$\min_{D_i} \sum_{i=1}^{M}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+ + \lambda\left(\sum_{i=1}^{M} D_i - D\right). \tag{7.32}$$

Let us write down the Karush-Kuhn-Tucker (KKT) conditions[9]:

$$\begin{aligned}\frac{d}{dD_j}\left\{\sum_{i=1}^{M}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+ + \lambda\left(\sum_{i=1}^{M} D_i - D\right)\right\} &= 0 \quad \forall j\\ \sum_{i=1}^{M} D_i - D &= 0\\ \lambda &\geq 0.\end{aligned} \tag{7.33}$$

[9] For nonlinear optimization problems, the Karush-Kuhn-Tucker (KKT) conditions are *necessary conditions* that a solution has to satisfy for being optimal. In some cases, the KKT conditions are also sufficient for optimality; this happens when the objective function is convex and the feasible set is convex too (convex inequality constraints and linear equality constraints). The system of equations corresponding to the KKT conditions is usually solved numerically, except in the few special cases where a closed-form solution can be derived analytically.
In the minimization problem considered here, the KKT are necessary and sufficient conditions for optimality. Besides, we will be able to solve them analytically.

Solving system (7.33) is complicated by the presence, in the objective function, of the plus sign in the subscript of the terms of the sum. Let us assume for the moment that $D_i \leq \sigma_i^2 \ \forall i$; in this case we have the system

$$\begin{array}{rcl} \frac{d}{dD_j}\left\{\sum_{i=1}^{M}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)+\lambda\left(\sum_{i=1}^{M}D_i-D\right)\right\} & = & 0 \quad \forall j \\ \sum_{i=1}^{M}D_i-D & = & 0 \\ \lambda & \geq & 0. \end{array} \tag{7.34}$$

The computation of the KKT conditions in (7.34) is now much easier:

$$\begin{array}{rcl} \frac{d}{dD_j}\left(-\frac{1}{2}\log D_j\right)+\lambda & = & 0 \quad \underset{(1)}{\longrightarrow} \quad D_j=\frac{1}{2\lambda}=\lambda', \quad \forall j \\ \sum_{i=1}^{M}D_i-D & = & 0 \quad \underset{(2)}{\longrightarrow} \quad D_j=\frac{D}{M} \\ \lambda' & \geq & 0. \end{array}$$

If $\frac{D}{M} \leq \sigma_i^2 \ \forall i$, then the solution of the minimization (7.31) is $D_i = \frac{D}{M}$ for each i, that means *distributing the distortion equally among the variables.* Note that this does not correspond to allocating the bits/symbol equally among the variables since $D_i \to R(D_i) = \frac{1}{2}\log\frac{\sigma_i}{D_i}$ (more bits are allocated to the r.v.'s with larger variance).
If instead for some i $\frac{D}{M} > \sigma_i^2$, it is straightforward to argue that allowing the random variables take a distortion D/M does not make sense. Indeed, when an admitted distortion $D_i = \sigma_i^2$ is achieved for a variable X_i, this means that we are assigning no bits to the random variable. Therefore, for the random variables X_i such that the distortion D/M exceeds the value of the variance σ_i^2, the best thing to do is to assign to them a distortion $D_i = \sigma_i^2$ and to reallocate the 'surplus' $(\frac{D}{M} - \sigma_i^2)$ uniformly among the remaining variables in order to reduce the bits/symbol for them (*compensation principle*). Then the optimal distortion distribution is

$$D_i = \begin{cases} \lambda & \text{if } \lambda < \sigma_i^2 \\ \sigma_i^2 & \text{if } \lambda \geq \sigma_i^2, \end{cases} \tag{7.35}$$

where λ satisfies $\sum_{i=1}^{M} D_i = D$.
The method described is a kind of *reverse water-filling* and is graphically

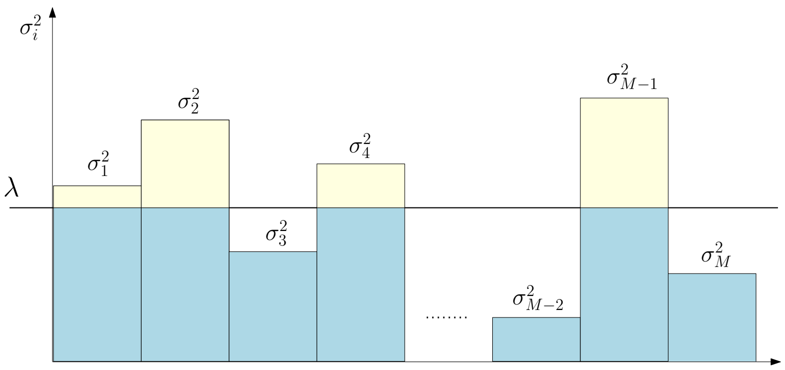


Figure 7.7: Reverse water-filling procedure for independent Gaussian random variables.

illustrate in Figure 7.7.
It is possible to prove that solution (7.35) is the same solution that we would have found by solving the initial set of KKT conditions in (7.33).
Then, the rate distortion function for M independent Gaussian sources with overall maximum distortion D is

$$R(D) = \sum_{i=1}^{M} \frac{1}{2} \log \frac{\sigma_i^2}{D_i}, \tag{7.36}$$

with $\{D_i\}_{i=1}^{M}$ satisfying (7.35).

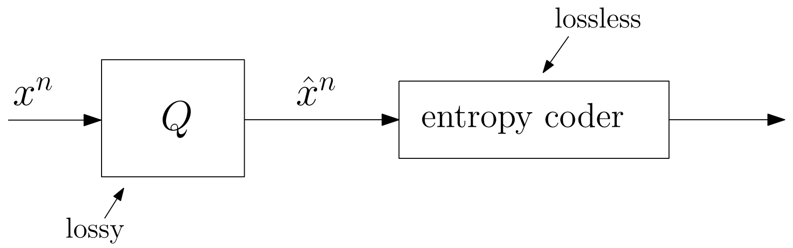


Figure 7.8: Quantization scheme.

7.2 Lossy Coding

7.2.1 The Encoding procedure in practice

The rate distortion theorem quantifies the trade-off between distortion and coding rate for lossy coding by means of the rate distortion function $R(D)$. For any source X and distortion measure D, $R(D)$ gives the minimum number of bits symbol required to reconstruct the source with a prescribed maximum distortion. As already pointed out in Section 7.1.1, we emphasize that, as it was the case with the source coding and the channel coding theorems, the values provided by the rate distortion function are 'fundamental limits': they can be achieved asymptotically and with increasing complexity of the encoding-decoding scheme (again, Shannon's scheme cannot be implemented).
In this section we search for the 'best' possible quantization procedure, that is the procedure which allows in practice to get as close as possible to $R(D)$. Specifically, we search for the set of reconstruction sequences $\{\hat{x}_i^n\}$ which satisfies the reconstruction distortion constraint. Figure 7.8 illustrates the idea. The encoder (Q) observes the source outputs $x^n \in \mathbb{R}^n$ (or $\mathcal{X}^n$) and maps them into representation sequences of length n, $\hat{x}^n \in \hat{\mathcal{X}}^n$. The quantization scheme should work on long blocks of source outputs (*vector quantization*). Indeed, similarly to what happened for the lossless source coding, quantizing together the random variables allows to reduce the rate even for memoryless sources (independent r.v.). The presence of the downstream entropy coder is due to the following reason: using any practical (suboptimum) quantization scheme, the bit rate at the output of Q does not correspond to the entropy of the output source, or equivalently the probability distribution of the encoded/assigned index is probably far from being uniform. Therefore, we can improve the compression efficiency through lossless coding, thus get-

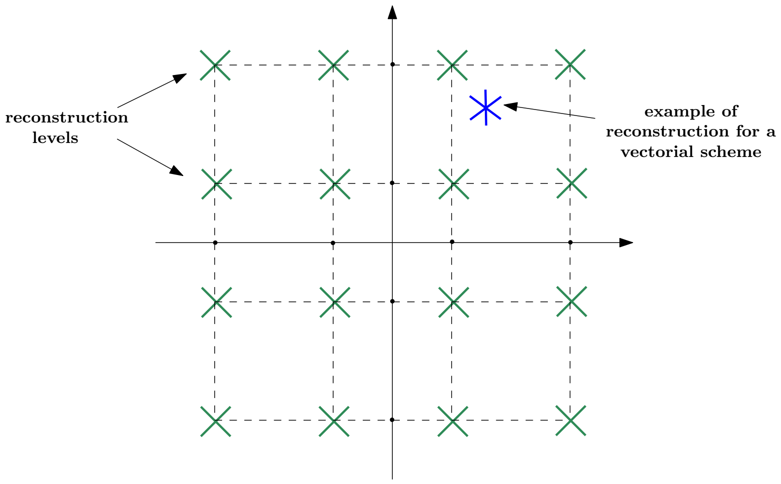


Figure 7.9: Scalar quantization: arrangement of the reconstruction points in a regular lattice (green crosses).

ting closer to $R(D)$[10].
The rest of this chapter is devoted to the study of the quantization process. Without any loss of generality, from now on we will refer to the continuous source case.

Quantization

Let $x^n \in \mathbb{R}^n$ denote a n-long vector of source outputs. In *scalar quantization* each single source output x_i is quantized into a number of levels which are later encoded into a binary sequence. In *vector quantization* instead the entire n-length vector of outputs is seen as a unique symbol to be quantized. A vector quantization scheme allows a much greater flexibility with respect to the scalar one, at the price of an increasing complexity. Let us see an example.

Suppose that we want to encode with a rate R. Assume for simplicity $n = 2$. Figure 7.9 shows how, through the scalar procedure, the reconstruction (quantized) levels in the $\mathcal{R}^2$ space are constrained to be disposed on a regular rectangular lattice and the only degrees of freedom are the quantization steps along the two axes. In general, we have $n2^R$ steps to set. Through the vector procedure, instead, we directly work in the $\mathcal{R}^2$ space and we can

[10] In the rate distortion theorem entropy coding is not necessary. Shannon proves that the distortion jointly encoding scheme reaches $R(D)$ which is the minimum. Then, it is as if the reconstructed sequences come out according to a uniform distribution.

put the reconstruction vectors wherever we want (e.g. the blue star). We have 2^{nR} points to set in general. Nevertheless, because of the lack of regularity, all the 2^{nR} points must be listed and for any output vector x^n all the 2^{nR} distances must be computed to find the closest reconstruction point, with a complexity which increases exponentially with n.

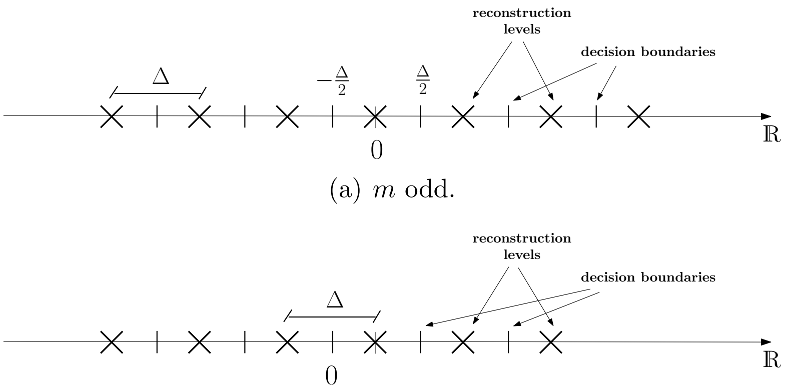
7.2.2 Scalar Quantization

Defining a scalar quantizer corresponds to define the set of *quantized or reconstruction levels* $\hat{x}_1, \hat{x}_2, ..., \hat{x}_m$ ($m = 2^R$) and the corresponding *decision regions* R_i, $i = 1, 2, ..., m$ (the partitioning of the space $\mathbb{R}$). Since the decision regions are intervals they are defined by means of the *decision boundaries* a_i, $i = 1, 2, ..., m$.

Uniform quantizer

The uniform quantizer is the simplest type of quantizer. In a uniform quantizer the spacing among the reconstruction points have the same size, that is the reconstruction levels are spaced evenly. Consequently, the decision boundaries are also spaced evenly and all the intervals have the same size except for the outer intervals. Then, a uniform quantizer is completely defined by the following parameters:
- levels: $m = 2^R$;
- quantization step: Δ.
Assuming that the source pdf is centered in the origin we have the two quantization schemes depicted in Figure 7.10, depending on the value of m (odd or even). Through a uniform scheme, once the number of levels and the quantization step are fixed, the reconstruction levels and the decision boundaries are univocally defined.
Figure 7.11 illustrates the *quantization function* $Q : \mathbb{R} \to \mathcal{C}$ (where $\mathcal{C}$ is a numerable subset of $\mathbb{R}$). For odd m, the quantizer is called a *midthread quantizer* since the axes cross the step of the quantization function in the middle of the tread. Similarly, when m is even we have a so called *midrise quantizer* (crossing occurs in the middle of the rise).

In the sequel, we study the design of a uniform quantizer first for a source having a uniform distribution, and later for a non uniformly distributed source.



(a) m odd.

(b) m even.

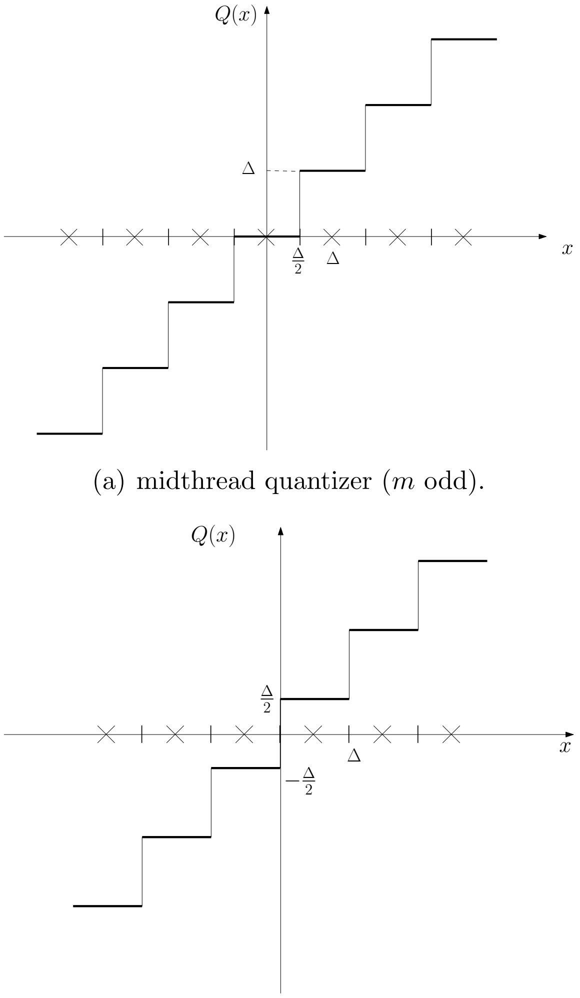
Figure 7.10: Uniform quantization: reconstruction levels and decision boundaries.

- *X uniform in [-A,A].*
  It is easy to argue that for this type of distribution the uniform quantizer is the most appropriate choice.
  Given m (number of reconstruction levels), we want to design the value of the parameter Δ which minimizes the distortion D, that is the quantization error/noise. Being the distribution confined in the interval $[-A, A]$ we deduce that $\Delta = \frac{2A}{m}$.
  The distortion (mean square quantization error) has the expression:

$$\begin{aligned} D &= E[(X - \hat{X})^2] \\ &= \int_{\mathbb{R}} (x - Q(x))^2 f_X(x) dx \\ &= \sum_{i=1}^{m} \int_{R_i} (x - \hat{x}_i)^2 f_X(x) dx \\ &\overset{(a)}{=} \sum_{i=1}^{m} \frac{1}{2A} \int_{R_i} (x - \hat{x}_i)^2 \\ &= \frac{m}{2A} \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} x^2 dx \\ &= \frac{1}{\Delta} \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} x^2 dx = \frac{\Delta^2}{12}, \end{aligned} \tag{7.37}$$

where each element i of the sum in (a) is the inertia moment centered on $\hat{x}_i$.

In this case, it is easy to see that the indexes obtained through the



(a) midthread quantizer (m odd).

(b) midrise quantizer (m even).

Figure 7.11: Quantization functions for a uniform quantizer.

encoding (outputs symbols) are uniformly distributed too. In this case entropy coding is useless.
Remembering that the variance of a source uniformly distributed in $[-A, A]$ is $A^2/3$, we can compute the signal to noise ratio as follows:

$$SNR = \frac{A^2/3}{\Delta^2/12} = m^2 = 2^{2R}. \tag{7.38}$$

Then, the scalar uniform quantization of a uniformly distributed source yields an $SNR = 6R$ dB, with a 6 dB increase for each additional bit[11].

- *Non uniform* X.
  In this situation there are some ranges of values in which it is more probable to have an observation. Then, we would like to increase the density of the reconstruction levels in the more populated zone of the distribution and use a sparser allocation in the other regions.
  This is not possible by using a uniform quantizer (the only parameter we can design is the spacing Δ). The question is then how to design the constant spacing Δ in order to achieve the minimum reconstruction error.
  Let us suppose m to be even (similar arguments hold if m is odd). We must compute:

$$\begin{aligned} D =& \sum_{i=1}^{m} \int_{R_i} (x - \hat{x})^2 f_X(x) dx \\ =& 2 \left\{ \underbrace{\sum_{i=1}^{m/2-1} \int_{(i-1)\Delta}^{i\Delta} \left(x - i\Delta + \frac{\Delta}{2} \right)^2 f_X(x) dx}_{\text{granular noise}} \right. \\ & \left. + \underbrace{\int_{(m/2-1)\Delta}^{\infty} \left(x - \frac{m}{2}\Delta + \frac{\Delta}{2} \right)^2 f_X(x) dx}_{\text{overload noise}} \right\}. \end{aligned}$$

The problem of finding the best Δ (minimizing D) must be solved numerically. However, it is interesting to point out the different contributions given by the two terms. Since Δ must have a finite value, through the quantization procedure we support only a limit range of the possible output values. This corresponds to clipping the output

[11] We remind that 6 dB is also the maximum growth of the SNR for added bit that we have for the Gaussian rate distortion function.

whenever the input exceeds the supported range. The second term of the sum is the clipping error, known as *overload noise.* Conversely, the error made for quantizing the values in the supported range is referred to as *granular noise* and is given by the sum at the first term. It is easy to see that decreasing Δ the contribution of the granular noise decreases, while the overload noise increases. Viceversa, if we increase Δ, the overload noise decreases at the price of an higher granular noise. The choice of Δ is then a tradeoff between these two types of noise, and designing the quantizer corresponds to finding the proper balance. Obviously, this balancing will depend on the to-be-quantized distribution, and specifically on how much it weighs the tail of the distribution with respect to the belly.

Example.
Numerical values for Δ obtained for a Gaussian distribution with $\sigma^2 = 1$:

1. $m = 2 \rightarrow \Delta_{opt} = 1.596 \quad SNR = 4.40dB$;
2. $m = 4 \rightarrow \Delta_{opt} = 0.996 \quad SNR = 9.24dB$;
3. $m = 8 \rightarrow \Delta_{opt} = 0,586 \quad SNR = 14.27dB$.

Note, that by increasing R by 1 bit/sample the SNR increases much less than 6 dB (limit value for the optimum quantizer).
It is possible to show that for a Laplacian distribution with the same m the value Δ_{opt} is larger. This is expected, being the tail of the Laplacian distribution much heavier than that of the Gaussian one.

Note: it's worth noting that using a uniform quantization for nonuniform distributions implies that the probabilities of the encoded output symbols (indexes) are not equiprobable and then in this case a gain can be obtained by means of entropy coding.

Non uniform quantizer

It is evident that the choice of a uniform quantizer for quantizing a non uniform source distribution is quite unnatural. At this purpose, it is surely more suited to resort to a non uniform quantizer which permits to exploit the local mass concentration of the probability distribution near the origin, where consequently the input is more likely to fall (see the example illustration in

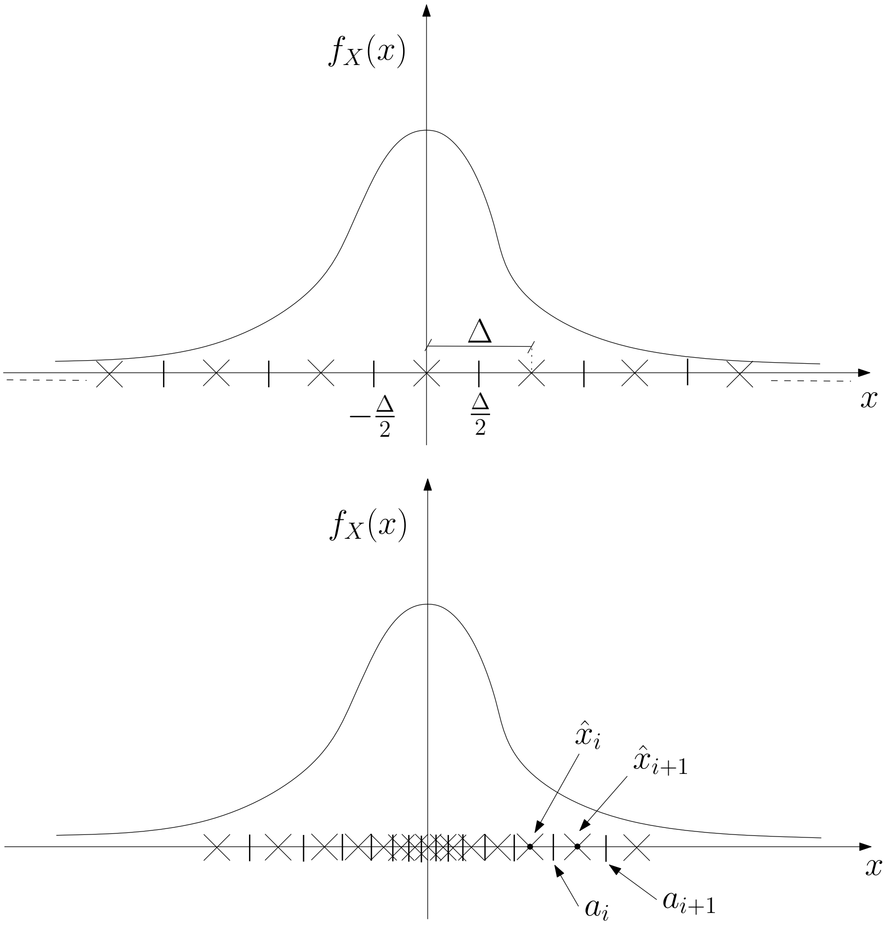


Figure 7.12: Example of uniform quantization (above) and non-uniform quantization (below) of a relatively peaked distribution.

Figure 7.12).

Again, suppose that the source is centered in the origin. With respect to the uniform quantizer, a nonuniform quantizer gives the designer much more freedom. Given the number of reconstruction levels m, the non uniform quantizer is defined by setting:

- reconstruction levels: $\hat{x}_i$, $i = 1, ..., m$;
- decision boundaries: a_i, $i = 0, ..., m$ (where $a_0 = -\infty$ and $a_m = +\infty$).

Hence, we have $2m - 1$ parameters (degrees of freedom) to set in such a way that the quantization error is minimized. It is easy to guess that, as in the example in Figure (7.12) (bottom), for non uniform sources, the optimum decision regions will have in general different sizes.

Max-Lloyd quantizer

In order to design the best nonuniform quantizer we have to search for the decision boundaries and the reconstruction levels that minimize the mean squared quantization error. Given the distortion error

$$\begin{aligned} D &= \sum_{i=1}^{m} \int_{R_i} (x - \hat{x}_i)^2 f_X(x) dx \\ &= \sum_{i=1}^{m} \int_{a_{i-1}}^{a_i} (x - \hat{x}_i)^2 f_X(x) dx, \end{aligned} \tag{7.39}$$

we have to minimize D by varying $(a_i)_{i=1}^{m-1}$ and $(\hat{x}_i)_{i=1}^{m}$.
Let us start by deriving the minimum with respect to the decision boundaries a_j, that is by computing $\frac{\partial D}{\partial a_j} = 0$ for $j = 1, ..., m-1$.

$$\begin{aligned} \frac{\partial D}{\partial a_j} &= \frac{\partial}{\partial a_j} \int_{a_{i-1}}^{a_i} \sum_{i=1}^{m} (x - \hat{x}_i)^2 f_X(x) dx \\ &= \frac{\partial}{\partial a_j} \int_{a_{j-1}}^{a_j} (x - \hat{x}_j)^2 f_X(x) dx + \frac{\partial}{\partial a_j} \int_{a_j}^{a_{j+1}} (x - \hat{x}_{j+1})^2 f_X(x) dx. \end{aligned} \tag{7.40}$$

Exploiting the following general relation:

$$\frac{\partial}{\partial k} \int_{\alpha}^{k} f(x) dx = f(k), \tag{7.41}$$

from (7.40) we get

$$\frac{\partial D}{\partial a_j} = -(a_j - \hat{x}_{j+1})^2 f_X(a_j) + (a_j - \hat{x}_j)^2 f_X(a_j). \tag{7.42}$$

Setting equation (7.42) to zero yields:

$$-(a_j - \hat{x}_{j+1})^2 + (a_j - \hat{x}_j)^2 = 0, \tag{7.43}$$

where we threw away the multiplicative constant $f_X(a_j)$ ($f_X(a_j) \neq 0$).
By exploiting the relation $a^2 - b^2 = (a + b)(a - b)$, after easy algebraic

manipulation we get:

$$(2a_j - (\hat{x}_j + \hat{x}_{j+1}))(\hat{x}_{j+1} - \hat{x}_j) = 0. \tag{7.44}$$

Since $(\hat{x}_{j+1} - \hat{x}_j) \neq 0$, we must have:

$$a_j = \frac{\hat{x}_j + \hat{x}_{j+1}}{2}, \quad \forall j. \tag{7.45}$$

Then, each decision boundary must be *the midpoint of the two neighboring reconstruction levels*. As a consequence, the reconstruction levels will not lie in the middle of the regions/intervals.

Let us now pass to the derivative with respect to reconstruction levels. We have to compute $\frac{\partial D}{\partial x_j} = 0$ for $j = 1, ..., m$.

$$\begin{aligned}\frac{\partial D}{\partial \hat{x}_j} &= \frac{\partial}{\partial \hat{x}_j} \sum_{i=1}^{m} \int_{a_{i-1}}^{a_i} \sum_{i=1}^{m} (x - \hat{x}_i)^2 f_X(x) dx \\ &= \frac{\partial}{\partial \hat{x}_j} \int_{a_{j-1}}^{a_j} (x - \hat{x}_j)^2 f_X(x) dx. \\ &= \int_{a_{j-1}}^{a_j} \left[\frac{\partial}{\partial \hat{x}_j} (x - \hat{x}_j)^2 \right] f_X(x) dx. \end{aligned} \tag{7.46}$$

Equating expression (7.46) to 0 yields

$$\cancel{2} \int_{a_{j-1}}^{a_j} (x - \hat{x}_j) f_X(x) dx = 0. \tag{7.47}$$

and then

$$\hat{x}_j = \frac{\int_{a_{j-1}}^{a_j} x f_X(x) dx}{\int_{a_{j-1}}^{a_j} f_X(x) dx} \quad \forall j = 1, ..., m. \tag{7.48}$$

By observing that $f_X(x|x \in R_j) = f_X(x) / \int_{a_{j-1}}^{a_j} f_X(x) dx$, we have

$$\begin{aligned}\hat{x}_j &= \int_{a_{j-1}}^{a_j} x f_X(x|x \in R_j) dx \\ &= E[X|X \in R_j]. \end{aligned} \tag{7.49}$$

Then, the output point for each quantization interval is the *centroid* of the probability density function in that interval.

To sum up, we have found the optimum boundaries $(a_i)_{i=1}^{m-1}$ expressed as a function of the reconstruction levels $(\hat{x}_i)_{i=1}^{m}$ and, in turn, the optimum reconstruction levels $(\hat{x}_i)_{i=1}^m$ as a function of $(a_i)_{i=1}^{m-1}$. Therefore, in order to find the decision boundaries and the reconstruction levels we have to employ an iterative procedure (Max Lloyd alghoritm): at first we choose the m reconstruction levels at random (or using some heuristic); then we calculate the boundaries and update the levels by computing the centroids:

$$\begin{cases} a_j = \frac{\hat{x}_j + \hat{x}_{j+1}}{2} & j = 1, ..., m-1 \\ \hat{x}_j = E[X|X \in R_j] & j = 1, ..., m. \end{cases} \tag{7.50}$$

The iterative procedure converges to a local minimum. However, the convergence to the absolute minimum of D is not guaranteed and depends on the choice of the initial conditions.

Entropy-constrained quantizer

With respect to the uniform quantizers, the nonuniform quantizer allow to define smaller step sizes in high probability regions and larger step sizes in low probability regions. This corresponds to 'equalize' the probability distribution. In this way, the gain achieved by means of the downstream entropy coder for the nonuniform quantization scheme is much less than that achieved in the uniform case. In order to understand this point we must stress that, in all the introduced quantization schemes, we have minimized D for a given number of reconstruction levels m (cardinality of the output alphabet). But what about the rate R which is the parameter we are interested in? Clearly, its value depends on the probability distribution of the output of the quantizer. When we have a uniform distribution and we quantize it uniformly, the distribution of the output indexes is uniform and then $R = \log m$. However, in general, when we deal with nonuniform distributions, the index distribution at the output of the quantizer is non uniform and then the effective rate is determined by the downstream entropy coder.

An alternative strategy is to design the quantizer *by fixing the rate at the output of the downstream entropy coder*, i.e. the entropy, rather than the output alphabet size (m).

The entropy of the quantizer output is:

$$H(Q) = -\sum_{i=1}^{m} P_i \log P_i, \tag{7.51}$$

where P_i is the probability that the input falls in the i-th quantization bin[12], that is

$$P_i = \int_{a_{i-1}}^{a_i} f_X(x)dx, \tag{7.52}$$

where a_{i-1} and a_i are the decision boundaries of the i-th decision region. Hence for a fixed rate R, the optimum decision boundaries a_i and reconstruction levels $\hat{x}_i$ can be obtained through the minimization of the distortion D subject to the constraint $R = H(Q)$.

Such a quantizer is called *Entropy constrained quantizer* (ECQ), and is the best nonuniform scalar quantizer. However the minimization with the additional constraint on $H(Q)$ is much more complex and must be solved numerically.

7.2.3 Vector Quantization

From the proof of the rate distortion theorem we argue that, similarly to what happened for lossless source coding, vector quantization gives advantages even when dealing with memoryless sources. However, block-wise quantization is even more important (actually essential) when we deal with memory sources. Below, we provide some examples to give an idea of the gain which derives by working with vector schemes with respect to scalar schemes, both for the memoryless and the memory case.

Since now the source outputs are quantized in blocks, the *reconstruction levels* are vectors in $\mathbb{R}^n$ (let n be the length of each block of symbols), namely $\hat{x}_1^n, \hat{x}_2^n, ..., \hat{x}_m^n$ ($m = 2^{nR}$)[13], and the *decision regions* R_i, $i = 1, 2, ..., m$, are partitions of the $\mathbb{R}^n$ space.

Vector quantization vs scalar quantization

Example (*<u>two Uniform Independent Sources</u>*).

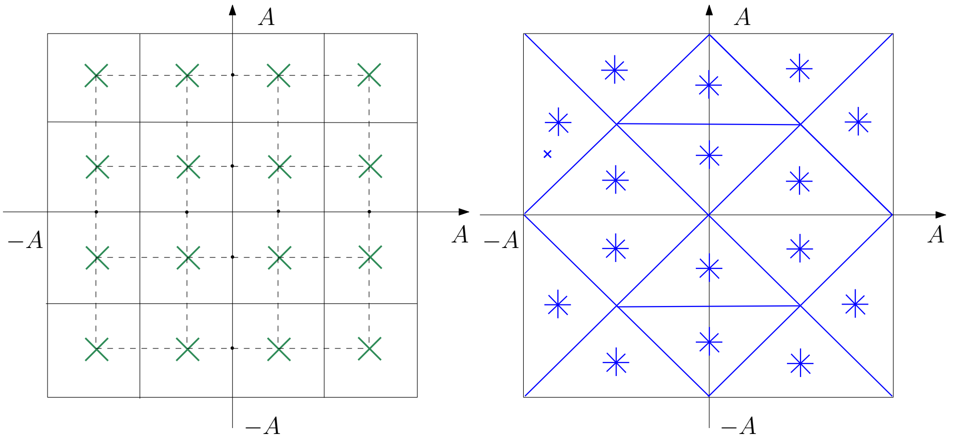
Let us consider the quantization of two memoryless sources X and Y both having uniform distribution in $[-A, A]$. To start with, let us suppose $R = 2$.

- ▶ *Scalar Quantization of the sources.*

  We can design a uniform quantizer by allocating the reconstruction lev-

[12] i.e. the probability that the output is $\hat{x}_i$.

[13] Sometimes, the vector notation $\hat{\vec{x}}_i$ (instead of x_i^n) is used to denote the reconstruction points.



(a) regular lattice (scalar quantization). (b) triangular tessellation (example of vector quantization).

Figure 7.13: Quantization of two Uniform Independent Sources.

els and the boundaries as in Figure 7.13(a). For uniform distributions it is easy to guess that the uniform quantization is also the optimal solution of the Max-Lloyd quantizer (and the ECQ).
Accordingly, we have

$$\begin{aligned} D &= \int\int_{\mathbb{R}^2} (\vec{x} - Q(\vec{x}))^2 f_{\vec{X}}(\vec{x}) d\vec{x} \\ &= \sum_{i=1}^{m} \int\int_{R_i} ||\vec{x} - \hat{\vec{x}}_i||^2 \frac{1}{4A^2} dx, \end{aligned} \tag{7.53}$$

where each term i of the sum is the *central moment of inertia* (c.m.i.) of the region R_i. Since $\hat{\vec{x}}_i$ [14] is the central point of each region R_i and the regions are all the same in shape and dimension, the contribution of each term of the sum is the same (the c.m.i is translation invariant). Then we have

$$D = \frac{m}{4A^2} I_{\square}, \tag{7.54}$$

where $I_{\square}$ is the central moment of inertia of a square having area $\frac{4A^2}{16}$.

- ▶ *Vector quantization*
  If we use a vector scheme we have much more freedom in the definition of the quantization regions, being them no more constrained to a rigid reticular structure as in the scalar case. Dealing with uniform distri-

[14] which here is a couple $(\hat{x}_{1_i}, \hat{x}_{2_i})$.

butions, we might think that the possibility of choosing the shape of the regions does not lead to a gain. However, the vector quantization gives an advantage in terms of distortion even in such a case. Indeed, we can suppose to use decision regions of different form (e.g. triangular regions, as in Figure 7.13(b). Using for instance hexagonal regions having the same area of the square regions we have that $I_{\hexagon} < I_{\square}$, where $I_{\hexagon}$ is the moment of inertia of the hexagonal region. This simple choice for the v.q.[15] already diminishes the distortion. According to the behavior of the c.m.i., the distortion gain would be even higher if we used multi-sided geometrical figure (with the same area) to cover the space, the minimum of I being attained by the sphere. However, in all these cases we have a boundary effect due to the fact that it is not possible to exactly cover a square domain through figures with more than 4 sides, as shown in the example in Figure 7.14 for the hexagonal regions. This effect becomes negligible when we have many reconstruction levels (fine quantization). Furthermore, we point out that by using geometrical figure with many-sides (> 6) a without-gap coverage of the internal space is not possible (think to the limit case of the sphere).

So far, we have considered a simple example of quantization of blocks of symbols of length 2. The gain of the v.q. with respect the s.q. increases when we quantize blocks consisting of more than 2 symbols. The reason is twofold:

- the gain in terms of the c.m.i. obtained by using n-dimensional hypersphere in place of n-dimensional hypercube grows with n;
- the coverage of the space is possible by means of geometrical figures with a larger number of sides and the boundary effect becomes negligible.

Note: by considering the limit case, that is when the length of the block n approaches ∞, the problem faced with in rate distortion can be seen as a problem of 'sphere covering'. The sphere covering problem consists in finding the minimum number of spheres through which is possible to cover a given space satisfying a condition on the maximum reconstruction distortion (maximum ray of the spheres).

In the example above we have considered the case of independent uniform sources. It is possible to show that the gain of the v.q. with respect the s.q. is higher when we deal with the quantization of independent non-uniform

[15] We use v.q. as the short for vector quantization, s.q. for scalar quantization.

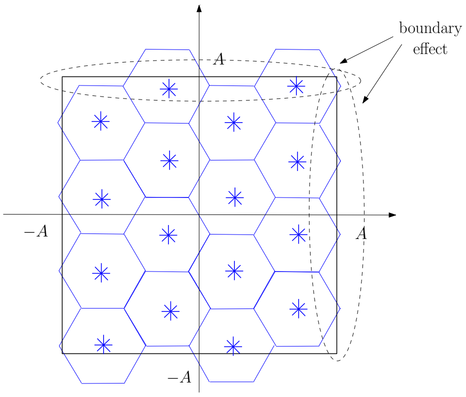


Figure 7.14: Hexagonal tessellation of the square domain: boundary effect.

sources, and especially of sources having peaked and tailed distributions. The reason is that, using the v.q., the freedom in the choice of the reconstruction levels allows to better exploit the greater concentration of the probability distribution in some areas with respect to others, thus reducing the quantization error.

We now consider an example of source with memory and show that in this case the gain derived by using the v.q. is really much stronger. We stress that the rate distortion theorem has been proved for the memoryless case. In the case of dependent sources the theorem should be rephrased by considering the entropy rate $\mathcal{H}$ in place of the entropy H. It is possible to prove that the theoretic limit value for the rate distortion $R(D)$ is lower than that of the memoryless case. This is not a surprise: for reconstructing the source with a given fixed distortion D the number of information bits required is less if we can exploit the dependence between subsequent outputs (correlation). Nevertheless, this is possible only by means of vector quantization schemes.

Example (*two Dependent Sources*).
Let us consider the problem of the quantization of two sources X and Y with joint distribution

$$f_{XY} = \begin{cases} \frac{1}{ab} & (X, Y) \in \mathcal{A} \\ 0 & (X, Y) \notin \mathcal{A}, \end{cases} \tag{7.55}$$

where $\mathcal{A}$ is the rectangular region of side lengths a and b depicted in Figure 7.15.

The correlation between X and Y makes necessary the resort to vector quantization. Indeed, any scalar quantization scheme looks at the marginal

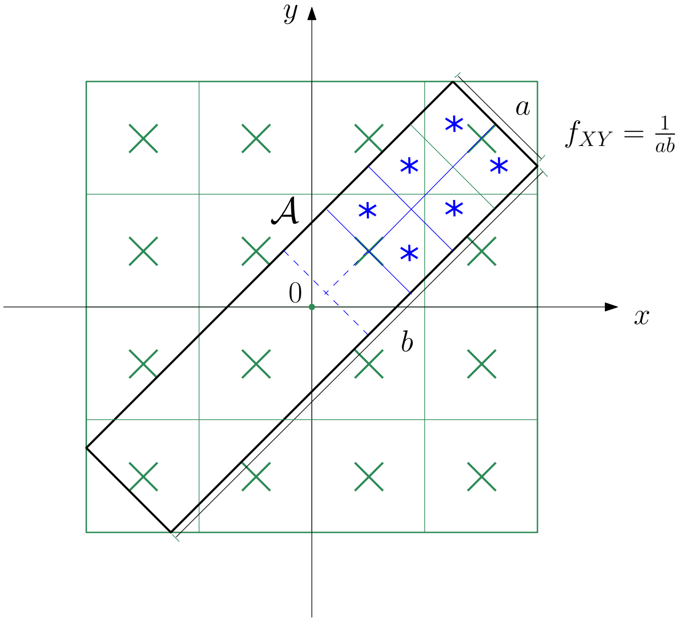


Figure 7.15: Example of two correlated random variables X and Y. The vector quantization (star tessellation, blue) is necessary since any scalar scheme (cross tessellation, green) leads to an unsatisfactory distribution of the reconstruction points.

distributions f_X and f_Y separately. Clearly, in this way, it places many reconstruction levels in regions in which the input couples never falls, causing a noticeable waste of resources. The situation is illustrated in Figure 7.15. Assuming for simplicity that the marginal distributions are not far from being uniform ($a << b$), or equivalently a fine quantization (high m), the optimum scalar solution for the quantization is to divide the space in m equally sized square regions. The distortion introduced is again

$$D = \frac{m}{4A^2} I_{\square}, \tag{7.56}$$

where now $I_{\square}$ is the c.m.i. of a square with area $(\frac{a+b}{\sqrt{2}})^2 \cdot \frac{1}{m}$.
Through a vector scheme, instead, by exploiting the correlation between the sources, we can place the decision regions only in $\mathcal{A}$ (see figure 7.15). In this way, for the same number of reconstruction levels, the area of each square region would be $\frac{ab}{N}$, which is much less than before. Consequently, the distortion is given by the c.m.i. $I_{\square}$ of a smaller square and then has a lower value.
This example shows that, for the case of source with memory, the gain achieved by the v.q. is more significant, since it derives from the *reduction of the size* of the regions and not only from the choice of a more suitable shape, as for the memoryless sources case.

The Linde-Buzo-Gray (LBG) algorithm

The Linde-Buzo-Gray algorithm is the generalization of the optimum Max Lloyd quantizer to vector quantization. The source output is grouped into n-length blocks and each resulting n dimensional vector is quantized as a unique symbol. In this way, the decision regions R_i can no longer be described as easily as in the case of the scalar quantization (where the regions were simply intervals!). However, in hindsight, the M-L quantizer is nothing else than a *minimum distance quantizer* [16]; then the update of the iterative algorithm can be generalized as follows:

$$\begin{cases} R_i = \{\vec{x} : ||\vec{x} - \hat{\vec{x}}_i|| < ||\vec{x} - \hat{\vec{x}}_j||\} & \forall j \neq i \\ \hat{\vec{x}}_i = E[\vec{X}|R_i] & \forall i. \end{cases} \tag{7.57}$$

The system defines the updating of the decision regions and reconstruction levels for the LBG algorithm.

[16] This is the meaning of placing the boundary points at the midpoints between the reconstruction levels.

The main practical drawbacks of the LBG quantization are:

1. the evaluation of the expected value requires the computation of a n dimensional integral. Besides, it requires the knowledge of the joint distribution $f_{\vec{X}}(\vec{x})$, which then should be estimated!

2. The convergence of the algorithm (in time and in space) strongly depends on the choice of the initial conditions, i.e. the initial placement of the quantization levels.

Clustering by K-means

For solving the problem of the estimation of $f_{\vec{X}}(\vec{x})$, Linde, Buzo and Gray propose to design the vector quantizer by using a clustering procedure, specifically the K-means algorithm.
Given a large *training set* of output vectors $\vec{x}_i$ from the source and an initial set of k reconstruction vectors $\hat{\vec{x}}_j$, $j = 1, ..., k$, we can partition the points $(\vec{x}_i)$ instead of splitting the space $\mathbb{R}^n$. We define each cluster C_j as follows:

$$C_j = \{\vec{x}_i : ||\vec{x}_i - \hat{\vec{x}}_j|| \leq ||\vec{x}_i - \hat{\vec{x}}_w||, \forall w \neq j\}. \tag{7.58}$$

In this way, once the clusters are defined, we can update the reconstruction vectors by simply evaluating the mean value of the points inside each cluster (without having to compute any integral!). Then, the new levels are

$$\hat{\vec{x}}_j = \frac{1}{|C_j|} \sum_{i:\vec{x}_i \in C_j} \vec{x}_i, \quad \forall j = 1, .., k. \tag{7.59}$$

At each step, a distortion contribution D_j is associated to cluster j,

$$D_j = \frac{1}{|C_j|} \sum_{i:\vec{x}_i \in C_j} ||\vec{x}_i - \hat{\vec{x}}_i||, \quad \forall j. \tag{7.60}$$

By iterating the procedure of cluster's definition, (7.58), and update of the centroids, (7.59), the algorithm converges to a local optimum. However, the solution found and the convergence speed strongly still depends on the initial distribution of the reconstruction vectors.

7.2.4 Avoiding VQ: the decorrelation procedure

We have seen that vector quantization procedure is necessary to approach the $R(D)$ curve, but is computationally heavy. Even using the LBG algorithm for the design of the quantizer, the real problem is that the number of

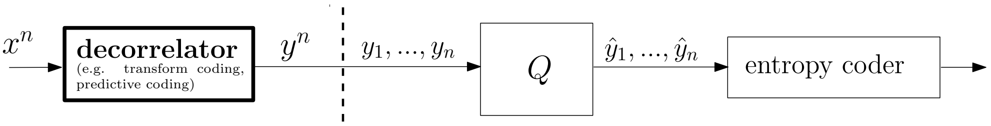

x^n
decorrelator
(e.g. transform coding, predictive coding)
y^n
$y_1, ..., y_n$
Q
$\hat{y}_1, ..., \hat{y}_n$
entropy coder

Figure 7.16: Avoiding vector quantization: the input sequence x^n is transformed by the decorrelator into a sequence y^n which has a lower correlation and then is more suited to be encoded scalarly.

reconstruction levels (centroids) we have to determine and store for a given rate is high, (2^{nR}), and above all grows exponentially with n. Besides, the quantization procedure requires to evaluate for each input 2^{nR} distances in order to determine the closest 'centroid'.

Then, for the memoryless case, in which the gain derived from the use of the v.q. is of minor importance, we are often content with the scalar quantization. Differently, for the case of sources with memory, the gain of the v.q. is significant and then we have to resort to clever tricks. The idea is to act on the output of the source $\vec{X}$ at the purpose of eliminating the dependence between subsequent symbols. The block at the beginning of the chain in Figure 7.16 implements a *transform based coder* or a *predicted coder* for decorrelating the outputs of the source. In this way, at the output of the decorrelator the scalar quantization can be applied without significant loss of performance.

Transform-based Coding

We search for a transformation A such that the new source $\vec{Y} = A\vec{X}$ shows no or little dependence between the variables.

We suppose for simplicity that $\vec{X}$ is a Gaussian vector with zero mean. Then, the multivariate pdf is

$$f_{\vec{X}}(\vec{x}) = \frac{1}{\sqrt{(2\pi)^n |C_x|}} e^{-\frac{\vec{x}^T C_x^{-1} \vec{x}}{2}} dx, \tag{7.61}$$

where $C_x = E[\vec{X} \cdot \vec{X}^T]$ is the covariance matrix ($C_x = \{C_{ij}\}_{i,j=1}^n$ with $C_{ij} = E[X_i X_j]$). Being C_x a *symmetric* and *positive definite* matrix, it admits an inverse matrix (C_x^{-1}).

Let us evaluate the behavior of the random vector after the transformation A is applied, i.e. the distribution of the output source $\vec{Y}$. Being the transformation linear, we know that $f_{\vec{Y}}(\vec{y})$ is still a multivariate Gaussian pdf with

$\vec{\mu}_y = 0$. The covariance matrix is

$$\begin{aligned} C_y =& E[\vec{Y} \cdot \vec{Y}^T] = E[A\vec{X} \cdot (A\vec{X})^T] = \\ =& E[A\vec{X} \cdot \vec{X}^T A^T] = AE[\vec{X} \cdot \vec{X}^T]A = AC_x A^T. \end{aligned} \tag{7.62}$$

Since C_x is positive definite, we can choose A in such a way that C_y is a diagonal matrix (Y_i independent r.v.). Being A the matrix which diagonalizes C_x (diagonalization matrix), the rows of A are formed by the eingevectors of C_x. Besides, since C_x is symmetric, there exists an orthonormal basis formed by the eigenvectors of C_x. Then, A is an orthonormal matrix ($A^T = A^{-1}$). As a consequence, the transformation does not change the entropy of the source and then does not lead to any loss of information. Indeed, the entropy of a Gaussian random vector $\vec{Y}$ ($h(\vec{Y})$) depends on $\vec{Y}$ only through the determinant of ${C_y}^{17}$, and we have

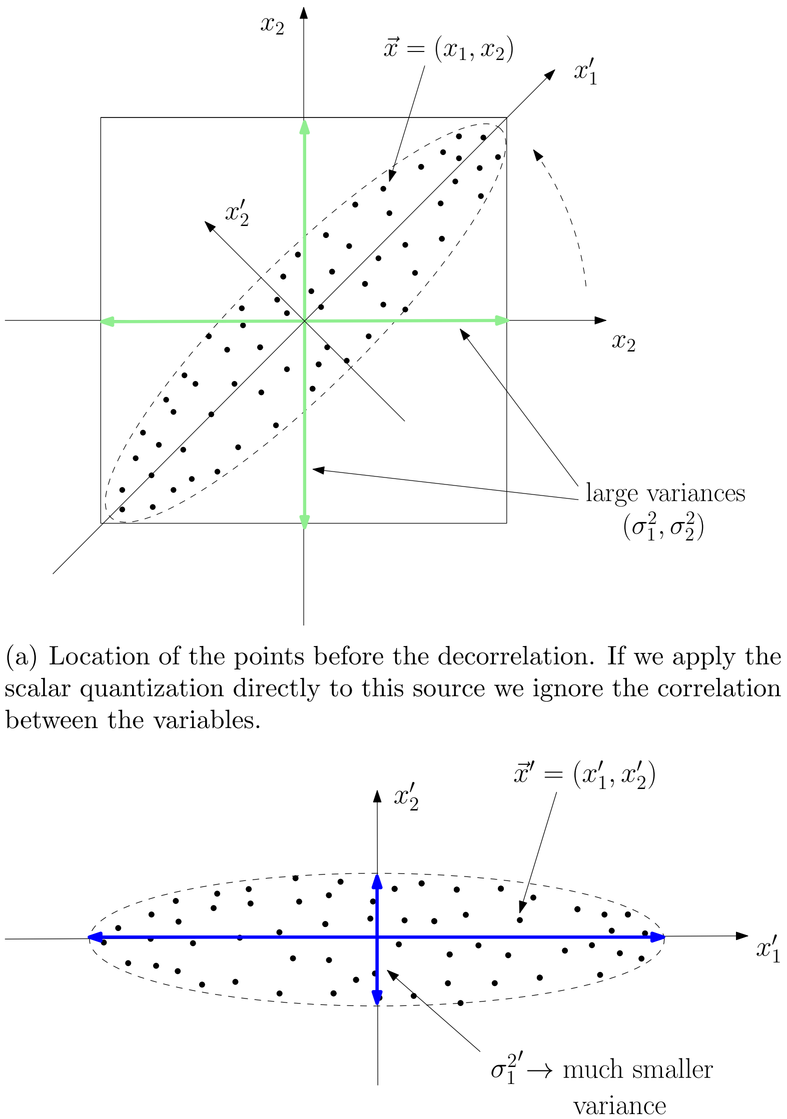
$$|C_y| = |AC_x A^T| = |A||C_x||A^{-1}| = |C_x|. \tag{7.63}$$

Then, it follows that $h(\vec{Y}) = h(\vec{X})$.
In this way, we can work on the source $\vec{Y}$ obtained at the output the transform block and only later go back to $\vec{X}$ ($\vec{X} = A^{-1}\vec{Y}$) without any loss.
Since $\vec{X}$ is a source with memory, we know that $h(\vec{X}) < \sum_i h(X_i)$ (remember that if the length of the vector n tends to infinity, $h(\vec{X}) \rightarrow n\mathcal{H}(\mathbb{X})$). On the contrary, the entropy of the decorrelated random vector $\vec{Y}$ which results from the diagonalization procedure can be computed as the sum of the entropies of the single r.v. Y_i. Indeed, from the resulting diagonal covariance matrix

$$C_y = \begin{pmatrix} \sigma_{y_1}^2 & 0 & \dots & \dots & 0 \\ 0 & \sigma_{y_1}^2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & \sigma_{y_n}^2 \end{pmatrix}. \tag{7.64}$$

it follows that $|C_y| = \prod_{i=1}^n \sigma_i^2$, and then

$$h(\vec{Y}) = \frac{1}{2}\log((2\pi e)^n|C_y|) = \sum_{i=1}^n \frac{1}{2}\log(2\pi e\sigma_i^2) = \sum_{i=1}^n h(Y_i). \tag{7.65}$$


x_2
$\vec{x} = (x_1, x_2)$
x'_1
x'_2
x_2
large variances
(σ_1^2, σ_2^2)
x'_2
$\vec{x}' = (x'_1, x'_2)$
x'_1
$\sigma_1^{2'} \rightarrow$ much smaller
variance

(a) Location of the points before the decorrelation. If we apply the scalar quantization directly to this source we ignore the correlation between the variables.

(b) Location of the points after the decorrelator. The variance of the random variables is greatly reduced as well as their correlation (which is approximately zero).

Figure 7.17: Variability of the couple of random variables before and after the decorrelation takes place.

At this point, we can apply the scalar quantization to the resulting source $\vec{Y}$ incurring only on the little gain loss we have in the memoryless case (discussed in Section 7.2.3).
Figure 7.17 illustrates the effect of the transformation for the case of two dependent Gaussian random variables X_1 and X_2. The point cloud in Figure 7.17(a) describes the density of the vectors $\vec{X} = (X_1, X_2)$, according to the bivariate Gaussian distribution. By looking at each variable separately, the ranges of variability are large ($\sigma^2_{x_1}$ and $\sigma^2_{x_2}$ are large). Therefore, as discussed in the previous sections, directly applying the scalar quantization to this source implies a waste of resources, yielding a high rate distortion $R(D)$[18]. Applying the transformation A to $\vec{X}$ corresponds to rotating the axes in such a way that the variability ranges for the variables are reduced (minimized), see Figure 7.17(b). It is already evident here (for $n = 2$) that decorrelating the variables corresponds to *compacting the energy* [19]. At the output of the transformation block we have independent Gaussian variables and then $R(D) = \sum_{i=1}^{n} R(D_i)$. At this point, we know that the best way for distributing the distortion between the variables is given by the Reverse Water Filling procedure, which allocates the bits to the random variables depending on their variance.

The Discrete Cosine Transform (DCT).

Given a source $\vec{X}$, the optimum transform for decorrelating the source samples is *Karhunen Loeve Transform (KLT).* The KLT is precisely the matrix A which diagonalizes the covariance matrix C_x. However, this implies that the KLT depends on the statistical properties of the source $\vec{X}$ and the covariance matrix C_x must be estimated for computing the KLT. Furthermore, for non stationary sources the estimation procedure must be periodically repeated.
As a consequence, computing the KLT is computationally very expensive. In practice, we need transforms that (although suboptimum) do not depend on the statistical properties of the data, but have a fixed analytical expression. The DCT (Discrete Cosine Transform) is one of the most popular transform employed in place of the KLT. For highly correlated Gaussian Markov sources, it has been theoretically showed that the DCT behaves like the KLT

[17] From the analysis of the previous chapter we remind that $h(\vec{Y}) = 1/2 \log((2\pi e)^n |C_y|)$.

[18] Remember that for a Gaussian r.v. $R(D) = \frac{1}{2} \log \frac{\sigma_x^2}{D}$.

[19] The overall energy is preserved by the transformation.

($KLT \approx DCT$). Then, in image compression applications[20], the DCT has the property of decorrelating the variables, even if, being *suboptimal*, some correlation remains among the transformed coefficients. This is the reason why many compression schemes (e.g. JPEG) work in the frequency domain: being the DCT coefficient almost decorrelated, scalar quantization can be applied with a negligible gain loss. The DCT transform compacts the energy into a small number of coefficients: the variance of the DCT coefficients is large at low frequency and decreases at high frequencies. In this way, due to the low variability of the coefficients in high frequency, through the procedure of Reverse Water Filling we can allocate 0 bits to them, that is discard them, introducing a very small distortion.

Predictive Coding

When the elements of the vector $\vec{X}$ are highly correlated (large amount of memory among the components) consecutive symbols will have similar values. Then, a possible approach to perform decorrelation is by means of 'prediction'. Using the output at the time instant $n-1$, i.e. X_{n-1}, as the prediction for the output at the subsequent time instant n (*Zero Order Prediction*), we can transmit only the 'novel' information brought by the output X_n, that is:

$$D_n = X_n - X_{n-1}. \tag{7.66}$$

In this way, the quantizer Q works on the symbols d_n ($d_n = x_n - x_{n-1}$) [21].

Quantizing D_n instead of X_n has many advantages which derive from the following properties:

✓ $\sigma_d^2 << \sigma_x^2$;
remember that the variance is the parameter which affects the number of bits we have to spend for the encoding with prescribed maximum distortion; specifically, a lower σ^2 corresponds to a lower rate distortion function $R(D)$.

[20] The source of an image is approximately a Markov source with high enough correlation.

[21] The symbols d_n can be seen as the output (at time n) of a new source $\vec{D}$ with reduced memory.

Proof.

$$\begin{aligned} E[D_n^2] &= E[(X_n - X_{n-1})^2] \\ &= \sigma_x^2 + \sigma_x^2 - 2E[X_n X_{n-1}] \\ &\overset{(a)}{=} 2\sigma_x^2 - 2\rho\sigma_x^2 = 2\sigma_x^2(1-\rho), \end{aligned} \tag{7.67}$$

where (a) follows from the definition of the correlation coefficient ρ which for a couple of r.v. X and Y has the expression

$$\rho = \frac{cov(XY)}{\sigma_X \sigma_Y} = \frac{E[XY]}{\sigma_X \sigma_Y}. \tag{7.68}$$

Due to the high correlation between X_n and X_{n-1}, ρ is close to 1 and then from (7.67) holds $\sigma_d^2 << \sigma_x^2$. □

✓ $\rho_d << \rho$;
the correlation (memory) between the new variables D_n is less then the correlation among the original source outputs X_n (as an example, if the source is a first order Markov process the symbols d_n obtained according to (7.66) are completely decorrelated).
Then, working on the symbols d_n (instead of on x_n), the loss incurred by using scalar quantization is much smaller.

However, it must be pointed out that the impact of the quantization of the symbols d_n on the original symbols x_n is different with respect to the case in which we directly quantize x_n.
In detail, the problem incurred by considering the differences in (7.66) is described in the following.

Coding/decoding scheme (Open loop)
At the first step the encoder transmits the first symbol x_1. Then, at the second step:
⇒ Transmitter side:

$$d_2 = x_2 - x_1 \underset{Q}{\longrightarrow} \hat{d}_2 = d_2 + q_2, \tag{7.69}$$

(q_i denotes the quantization error at step i).
⇒ Receiver side: knowing x_1 and having received $\hat{d}_2$,

$$\hat{x}_2 = x_1 + \hat{d}_2 = x_1 + d_2 + q_2 = x_2 + q_2. \tag{7.70}$$

At the third step:

⇒ Transmitter side:

$$d_3 = x_3 - x_2 \underset{Q}{\longrightarrow} \hat{d}_3 = d_3 + q_3. \tag{7.71}$$

⇒ Receiver side: knowing only the quantized version of x_2,

$$\hat{x}_3 = \hat{x}_2 + \hat{d}_3 = x_2 + q_2 + q_3. \tag{7.72}$$

Proceeding in this way, the n-th decoded symbol is

$$\hat{x}_n = x_n + \sum_{i=2}^{n} q_i. \tag{7.73}$$

It is evident that this encoding scheme leads to a *propagation of the error* (or *drift*) at the receiver.

Then, for avoiding error propagation at the receiver, the encoder must employ a *closed loop encoding*, by computing the differences with respect to the quantized value at the previous step, i.e. $D_n = X_n - \hat{X}_{n-1}$. In the following, we analyze the scheme in detail.

Coding/Decoding scheme (Closed Loop)

In order to avoid the propagation of the decoding error, the coder must base the prediction on the quantized values of the source symbols instead than the original ones. In fact, defining the differences with respect to the quantized values, that is $d_n = x_n - \hat{x}_{n-1}$, at the third step we would have:

⇒ Transmitter side:

$$d_3 = x_3 - \hat{x}_2 \underset{Q}{\longrightarrow} \hat{d}_3 = d_3 + q_3. \tag{7.74}$$

⇒ Receiver side:

$$\hat{x}_3 = \hat{x}_2 + \hat{d}_3 = \hat{x}_2 + d_3 + q_3 = x_3 + q_3. \tag{7.75}$$

The n-th decoded symbol is now

$$\hat{x}_n = x_n + q_n, \tag{7.76}$$

thus avoiding that quantization errors are accumulated. Figure 7.18 illustrates the closed loop encoding scheme and the corresponding decoding procedure.

Note: we have considered the simplest type of predictor, i.e. the *Zero Or-*

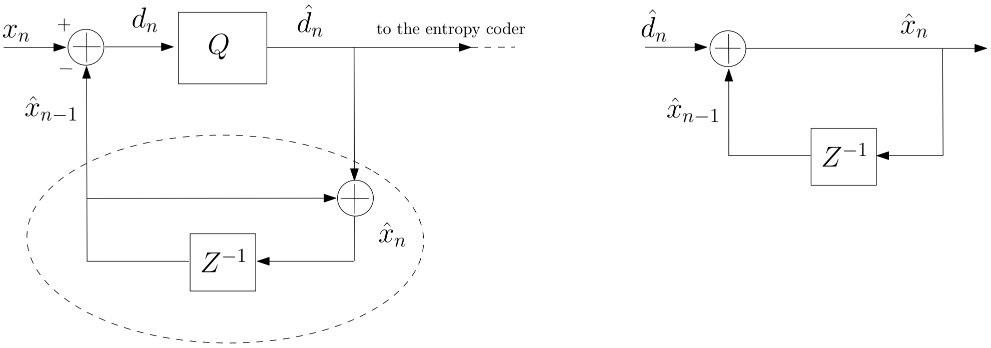


Figure 7.18: Predictive coding scheme. Closed loop encoder (on the left) and decoder (on the right).

der Predictor. More efficient prediction schemes can be obtained by using a linear combination of a certain number of source outputs (FIR filter).