HARDWARE AND DEVICE-TO-DEVICE COMMUNICATION SECURITY

Notes about that

Fabio Lorenzato

Contents

Part I

Wireless

Chapter 1

Digital Communication

This first section is all about how to convert and transmit some signal.

1.1 Introduction

The goal of communication is to transmit some kind of data form a sender to a receiver. In order to do so, the physical layer defines the means of transmitting a stream of **raw bits** over a physical data link, which connects those two nodes.
Data is transmitted in the form of **signals**, which are a physical representation of the data. The signal is transmitted over a **channel**, which is the transmission medium that connects the sender and receiver. This can be both wired or wireless.
Whereas with wired channels, checking the device connected to the channel is easier to implement, with wireless ones security is a major when travelling in the channel. This is for may reasons:

- No inherent protection is applied to the channel(it is replaced by a logical association)

  sending and receiving messages do not need physical access to the network infrastructure

- the communication is in broadcast, which is intrinsic of radio nature.

  Transmission can be overheard by anyone in range(which can be quite big, depending on the situation), and anyone can generate a transmission, for example by jamming nearby transmissions.

As a result:

- Eavesdropping is easy
- Injecting fake messages into the communication in easy
- replaying previously recorded messages is easy(*meaconing*). This is actually very dangerous for gps positioning, so it is also a security concern.
- illegitimate access to the network and its services is easy
- Denial of service attacks are easy, achieved by jamming the channel.

1.2 Digital Communication System

The digital communication system in characterized by three sections:

- the **user section**, which consist in the transmitter and the receiver, that want to communicate.
- the **interface section**, which is the interface to conveying the signal from the user to the analog channel. It also transforms bits to analog signal, compressing and encoding them, also associating bits to signal waveforms, to transform bits to analog signal.
- the **channel section**, which is the physical medium, that can only propagate analog waveforms. In the end, we want to transmit digital signal but we are forced to use analog ones.

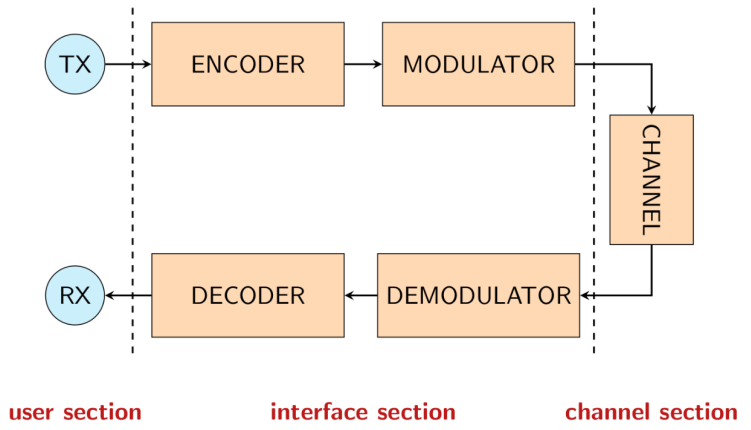


Figure 1.1: Digital Communication System

1.2.1 The transmitter chain

The transmitter chain is the part of the system that takes the digital signal, or an analog one converted to digital, and converts it to an analog signal, that can be transmitted over the channel.
It is basically composed by two parts. The first one being an **encoder**, which can limit the amount of bits transmitted(*source encoding*), and/or make the transmitted sequence more robust to errors(*channel encoding*).
The second one is the **modulator**, which is the part of the system that takes the digital signal and converts it to an analog one to transmit it over the channel.

1.2.2 The channel

The channel is the physical medium that transfers bits from interface to interface, from the sender to the receiver. Its operation is affected by different types of disturbances such as:

- frequency-domain distortion
- wireless fading

- additive noise
- impulsive noise
- interference from other frequency channels (interchannel interference)
- interference from the same frequency channel (cochannel interference)
- Intentional interference

1.2.3 The receiver chain

The receiver chain is the part of the system that takes the analog signal from the channel and converts it to a digital signal, that can be processed by the user.
It is composed by the dual counterpart of the transmitter chain, the **demodulator** and the **decoder**.
The demodulator takes the analog signal and converts it to a sequence of samples that can be processed by the decoder.
The decoder takes the sequence of samples and converts it to a digital signal. It implements *channel decoding*, to correct errors, and *source decoding*, to recover the original message.

1.3 Signal representation and Processing

> A **signal** is a (mathematical) function that conveys information about a phenomenon.

Basically, any quantity that varies over space or time can be used to represent a informations, allowing to describe the evolution of physical quantities over time(voltages, currents, ...).
Its mathematical representation is therefore a function of real variable (time) taking real or complex(more than one) values.
We will be mostly focused on Electromagnetic Signals (e.g. voltage), but the general concepts can be applied to any kind of signal

1.3.1 Energy of a signal

The energy of a signal is the integral of the squared modulus of the signal itself.

$$E(x) = \int_{-\infty}^{\infty} |x(t)|^2 dt \tag{1.1}$$

As we can see , the energy is a scalar value, and the whole function is made positive by the squared modulus.
A signal with a very large amplitude, over time, will have a very high energy, while a signal which assumes values close to zero will have a very low energy, being a very weak signal.
Furthermore, we can note that the more distant is the signal from the origin, the larger the energy.

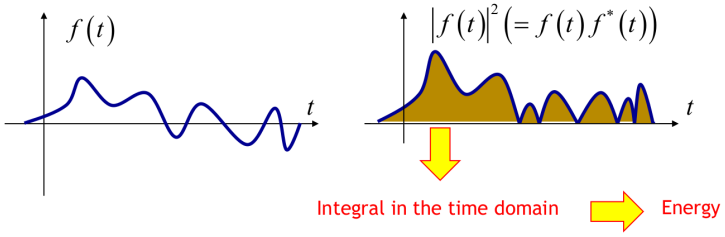


Figure 1.2: Energy of a signal

1.3.2 Power of a signal

When we refer to power we can refer to the **instantaneous power** of a signal, which is just the square module of a signal

$$P(x) = |x(t)|^2 \tag{1.2}$$

but much more commonly we refer to the average power of a signal, which is the time average of the instantaneous power of the whole signal.

$$P(x) = \lim_{a \to \infty} \frac{1}{2a} \int_{-a}^{a} |x(t)|^2 dt \tag{1.3}$$

This a again a scalar value.

1.3.3 Signal Representation

To analyze and process the signals, it is necessary to adequately represent them, and the definition of signals as "time functions" is NOT effective for many applications, for many reasons.
Generally, signals can become very complicated depending on our communication system, and we want different ways of representing them, to make them easier to process.
For instance, we can represent a signal as a sum of elementary signals, thanks to the scalar product of the signal with a basis of the space of signals.

The scalar product between signals is a scalar value, which is a measure of the similarity among signals.
If two function are quite similar we will get a large number. It it is zero, they are said to be orthogonal.

$$\langle x, y \rangle = \langle x(t), y(t) \rangle = \int_{-\infty}^{\infty} x(t) y^*(t) dt \tag{1.4}$$

So, if we have a set of elementary signals $w_1(t), w_2(t), \ldots, w_m(t)$, we write the signal $x(t)$ as a linear combination of the elementary signals:

$$x(t) = \sum_{i=1}^{m} \alpha_i w_i(t) \tag{1.5}$$

where α_i are the coefficients of the linear combination $\alpha_i = \langle x(t), w_i(t) \rangle$.

In a more down to hearth way, the coefficient α_i allows us to understand how much each individual signal is similar to any other elementary signal we are considering, and because the scalar product is higher for similar signals, we can understand how much each elementary signal is contributing to the whole signal.

Furthermore, by adjusting the coefficient, we are able to create a whole different signal using the same elementary signals.

A common example: In Phase and Quadrature components representation

Lets consider a very simple basis, or a set of elementary signals, which is actually more important that many other ones:

- the **in-phase** signal, which, in this case, is a cosine function $w_1(t) = cos(2\pi f_o t)$
- the **quadrature** signal, which, in this case, is a sine function $w_2(t) = sin(2\pi f_o t)$

where f_o is the frequency, in Hz, of the signal.
We can write any signal as a linear combination of these two signals, just by adjusting the coefficients:

$$x(t) = x_1 cos(2\pi f_o t) + x_2 sin(2\pi f_o t) \tag{1.6}$$

where x(t) is the signal we want to represent, and x_1 and x_2 are the coefficients of the linear combination.

A very simple representation of this complex signal is obtainable by representing each signal as an axes in a complex plane, for example in figure 1.3 the x-axis is the in-phase signal, and the y-axis is the quadrature signal.
Each signal can be represented as a point in the complex plane, because the distance from the origin signal(*axis*) is the amplitude of the signal.
This kind of representation is called the **In-phase and Quadrature** representation, or **I/Q** representation.
For example, choosing a point close to the x-axis, we are choosing a signal with a very low quadrature component, and a very high in-phase component.Furthermore, a set of those different points is called a *constellation*

1.3.4 Fourier Analysis

Lets consider a signal with base the complex exponential functions

$$e^{j2\pi \frac{n}{T} t} = cos(2\pi \frac{n}{T} t) + jsin(2\pi \frac{n}{T} t) \tag{1.7}$$

It is actually characterized by a frequency $f_n = \frac{n}{T}$, where T is the period of the signal. The higher the frequency, the more oscillations we will have in the same time interval. In this function they have both the same frequency.

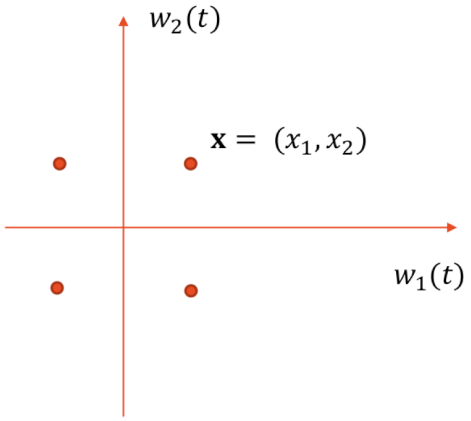


Figure 1.3: Some points that represent some signals in the I/Q representation

We can use that function as a basis to decompose a signal, again. This is because it is possible to generate an infinite set of functions

$$w_n(t) = \frac{1}{\sqrt{T}} e^{j\frac{2\pi}{T}nt} \tag{1.8}$$

with $-T/2 \leq t \leq T/2$, each associated with a frequency. That can be used ad a complete basis for all the signals limited in $[-T/2, T/2]$ or periodic.
For example, we can write a signal as a linear combination of these functions:

$$x(t) = \frac{1}{\sqrt{T}} \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi}{T}nt} \tag{1.9}$$

where c_n are the coefficients of the linear combination $c_n = \langle x(t), w_n(t) \rangle$.
Each one of those coefficients is a measure of how much each frequency f_n, of the n-th sinusoid(the shape of equation 1.7) is present in the signal $x(t)$.
Lets now take a look at picture 1.4. We can see that the coefficients are higher when the

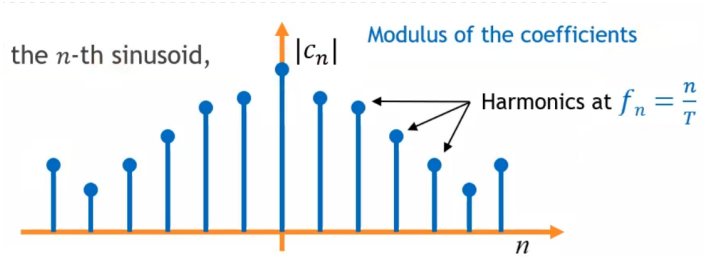


Figure 1.4: Plot of the coefficients of equation 1.9

frequence is very small, so the signal 1.9 is mostly composed by large components of low frequency.
This whole concept is called **Fourier Analysis**, or frequency analysis, which allows to decompose a signal into a set of frequencies.

TLDR: I can build a signal trough a combination of frequency components.
The coefficients of this frequency components are the measure of how much each frequency is present in the signal.

Now we just need to expand it to any signal and any frequency(a continuous frequency domain). By doing so we can derive the definition of the **Fourier Transform** of a signal $x(t)$:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{1.10}$$

where $X(f)$ is the Fourier Transform of the signal $x(t)$, and f is the frequency.
The Fourier transform is equivalent to a scalar product between the signal and the complex exponential function at a given frequency f. This means that each of the values of the Fourier Transform is a measure of how much the frequency f is present in the signal $x(t)$.
Furthermore, trough the inverse of equation 1.10

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{1.11}$$

we can write again a signal $x(t)$ as a linear combination of the complex exponential functions , which represents the frequency components of the signal, weighted by the Fourier Transform.

The Fourier Transform $X(F)$ indicates the "weight" of each frequency component(sinusoidal component at a given frequency f) in the signal $x(t)$.
The inverse Fourier Transform $x(t)$ tells us we can decompose any signal into frequency components(sinusoidal components at a given frequency f).

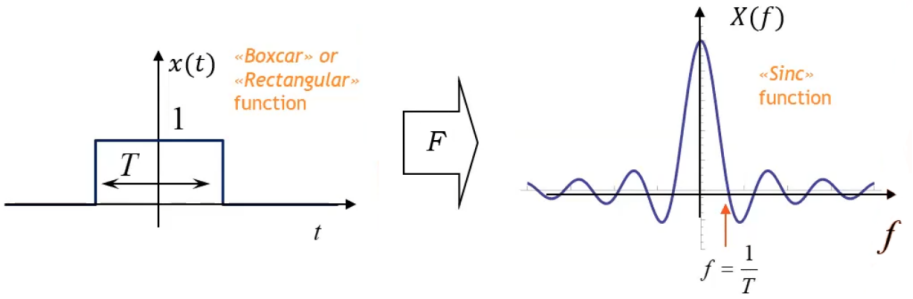


Figure 1.5: Fourier Transform of a square function

With that in mind, take a look at figure 1.5. It represents a rectangular signal (a signal that is 1 for a certain time, and 0 for the rest of the time), and its Fourier Transform, which tells us the frequency components of the signal.
From that, we can see that the Fourier Transform is mostly composed by low frequency components, because values closer to zero are higher. That is because in the constant part of the signal has a sinusoidal component that constant.

To wrap it up, for each signal we have a **spectral representation**. And for each operation over a signal, there are equivalents effects in the frequency domain. Furthermore, a signal that has finite duration in time, has a infinite support in the frequency domain.

1.3.5 Bandwidth

The bandwidth is the **interval of frequencies** that a signal occupies.
If we consider a signal $x(t)$, we can define the bandwidth as the interval of frequencies where the Fourier Transform $X(f)$ is different from zero.
Signals have often infinite support over the frequency domain over a finite duration, but many of them are characterized by a quasi-null(finite) spectrum outside a certain interval of frequencies(the main lobes of the spectrum).
For this reason, we usually consider the bandwidth around half of the frequency spectrum of the signal, as shown in figure 1.6(for example 3dB bandwidth, or half power bandwidth).

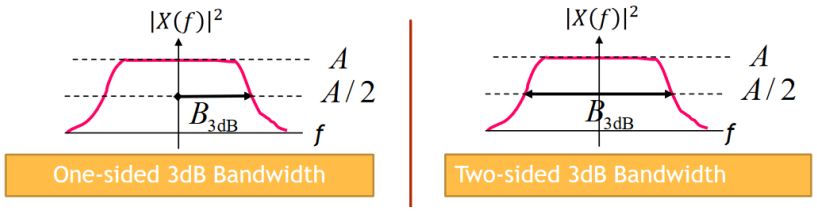


Figure 1.6: Bandwidth of a signal

Bandwidth in linear systems

A **system** is a set of operations applied to signals.
The relationship between the bandwidth of the input signal and the bandwidth of a system is usually very important. In fact, when a system is used to pass or remove particular frequencies of a signal, it can be regarded as a system.
We can associate a bandwidth to a system, specifically a **linear-time invariant** system, with a **frequency response** $H(f)$.This mean that we can associate a bandwidth to a system, making us able to compute a new bandwidth $Y(f)$ by combining together the bandwidth $X(f)$ of the input signal and the frequency response $H(f)$ of the system($Y(f) = X(f)H(f)$ in formulas).
This concept can be represented graphically very easily, like in figure 1.7. If the result of the combination of the input signal and the sequence of operation of the system is a signal with a bandwidth $Y(f)$. If the bandwidth of the linear system is larger than the origin signal, the signal pass trough smoothly($Y(f) \approx X(f)$).
However, if the bandwidth of the signal is larger than the bandwidth of the system, the signal will be cutted off($Y(f) \neq H(f)$).

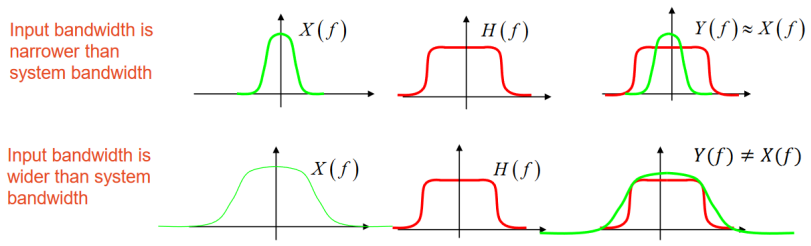


Figure 1.7: Bandwidth of a system

1.3.6 Filters

A filter is a system used to model desired and undesired effects over a signal.
It is usually used to remove undesired frequency components from a signal, but overall can be used to:

- share the wireless medium
- model the spectrum of a signal over the channel
- mitigate undesired effects over a signal trough equalizers

1.3.7 Signal modulation

Signal modulation is the process of multiplying a signal by a sinusoidal function, resulting in a **frequency shift**.

$$y(t) = x(t) \cdot cos(2\pi f_0 t) \tag{1.12}$$

This is possible because

$$F(x(t) \cdot cos(2\pi f_0 t)) = \frac{1}{2}[X(f - f_0) + X(f + f_0)] \tag{1.13}$$

where X is the frequency domain representation.
We can see that as a result of the modulation, the spectrum of the signal is shifted around

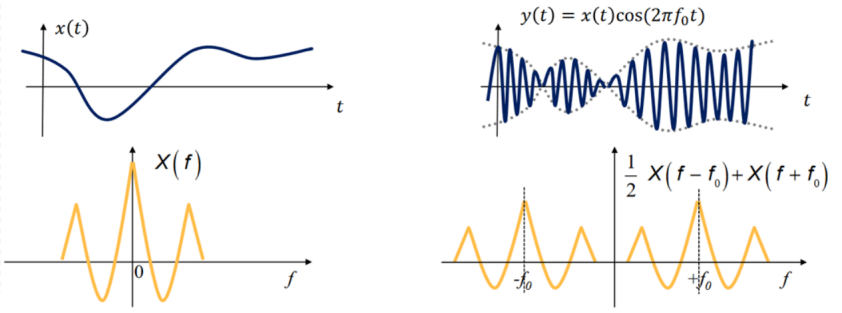


Figure 1.8: Graphical representation of the modulation

the frequency f_0, as shown in figure 1.8.

1.3.8 Signal demodulation

When modulating a signal, we alter it a bit, centering it around the frequency f_0, shifting the spectrum of the signal. The effect of this operation is not trivial.
To recover the original signal, we need to multiply the modulated signal by a sinusoidal function at the same frequency f_0 as the one used for the modulation. This allows us to shift the spectrum back to the original position.
This operation is called **demodulation**.
A given modulated signal $Y(f)$

$$Y(f) = \frac{A}{2}[X(f - f_0) + X(f + f_0)] \tag{1.14}$$

shown in figure 1.9 can be demodulated by multiplying it by the same sinusoidal function

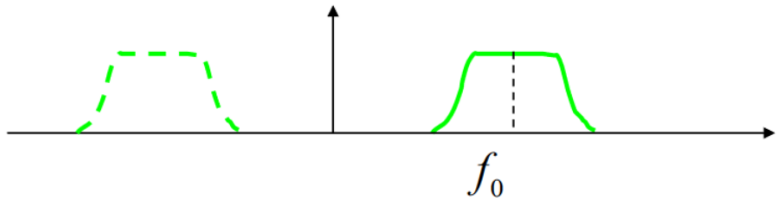


Figure 1.9: A modulated signal at frequency f_0

used for the modulation

$$Y'(f) = Y(f) \cdot cos(2\pi f_0 t) = \frac{A}{2}X(f) + \frac{A}{4}[X(f - 2f_0) + X(f + 2f_0)] \tag{1.15}$$

shown in figure 1.10
This doesn't allow us to recover the original spectrum of the signal. That's why we need

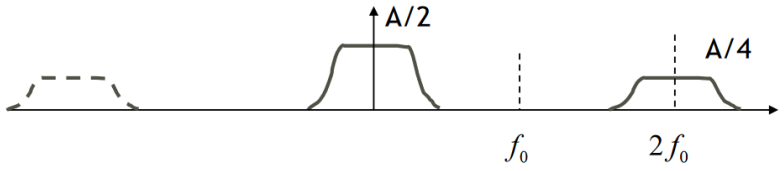


Figure 1.10: A demodulated signal at frequency f_0

to use a **low-pass filter** to remove the frequency components at $2f_0$ and its symmetrical counterpart, as shown in figure 1.11.

1.3.9 Frequency Multiplexing(FDM)

Modulation and demodulation allows multiple wireless communication systems to coexist at different frequencies.
For example, if i want to transmit different signals with overlapping bandwidths, i can simply modulate each signal at a different frequency, and then transmit them all together, as shown in figure 1.12.

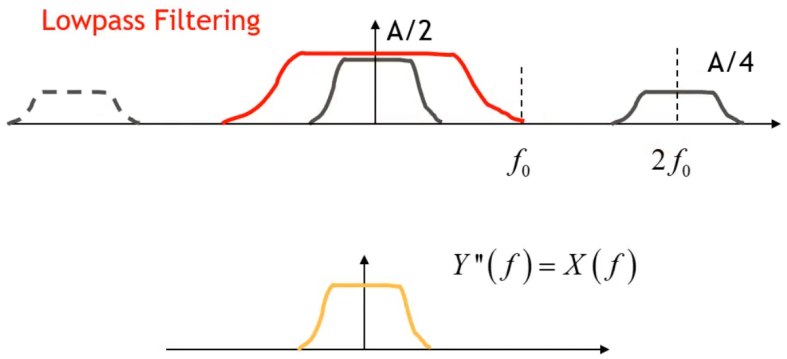


Figure 1.11: A demodulated signal at frequency f_0 after a low-pass filter

Once received the signal, each of those signals can be demodulated by multiplying it by the same function at the same frequency used for the modulation.

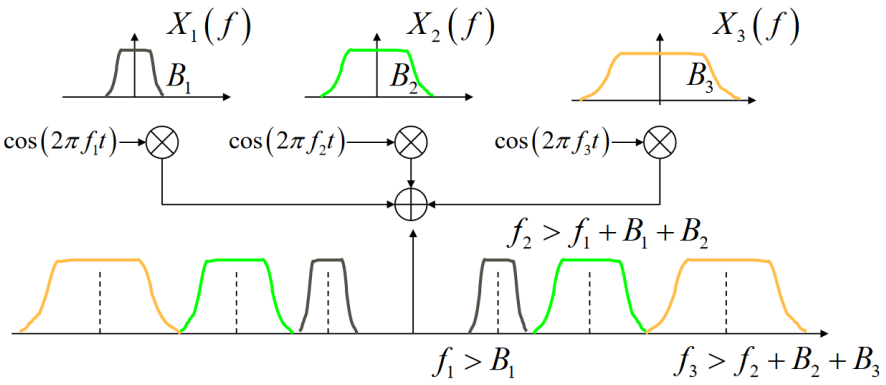


Figure 1.12: Frequency Multiplexing

1.3.10 Analog-to-Digital Conversion

We can now deal with signals, but we still have to convey information.
The informations can be both **analog** or **digital**. Usually, transmitting digital information is ideal, because it has some advantages, such as error detection and correction, and the possibility to compress the information. On the other hand, we still have to convert digital information to analog information to be transmitted over the channel, after converting it to a stream of bits.
Once the signal is received, it has to be converted back to digital information. To do so, first of all the signal it has to be sampled, which can be a lossless operation if the sampling frequency is high enough.
After the sampling, the sample has to be quantized, which is the process of converting the amplitude of the sample to a digital value at discrete times(because it is a continuous time function, which would require an infinite number of digits to represent). Each of those values is associated with a given amplitude, which is associated to a number, which eventually is converted into binary digits. We can also observe that quantization is a lossy operation by

definition.
At the end of the fair, a sequence of bits is obtained, which can be transmitted over the channel.

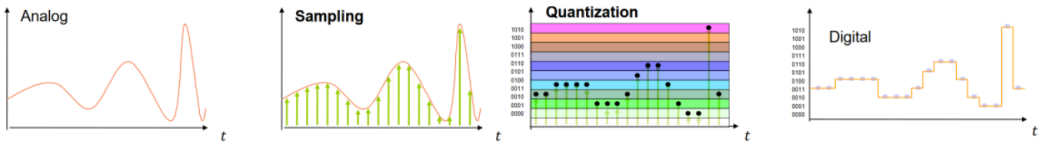


Figure 1.13: Analog-to-Digital Conversion

Sampling theorem

As previously stated, the sampling operation can be lossless if the sampling frequency is high enough. This is because of the **Nyquist sampling theorem**, which states that a signal can be perfectly reconstructed from its samples if the sampling frequency is at least twice the bandwidth of the signal.

$$f_c = \frac{1}{T_c} > 2B \rightarrow T_c < \frac{1}{2B} \tag{1.16}$$

where f_c is the sampling frequency, T_c is the sampling period, and B is the bandwidth.

1.4 Signal Transmission and Reception

Now that we know how a signal can be represented and processed, we can start to think how each component of a communication system can be designed.

1.4.1 Digital Modulations

The end goal is to have a reliable communication system, which can transmit and receive information. As such, a important design choose is the signal waveform to transmit.
The **modulator** is the component of the system that takes the digital information and modulates it to a signal that can be transmitted over the channel. The demodulator component just does the opposite, taking the signal and converting it back to digital information.

> **Modulation** is the process of varying one or more properties of a periodic waveform, called the **carrier**, with a modulating signal that typically contains information to be transmitted.

This process is necessary not only to cope with the analog channel, but also to allow multiple communication systems(which means different signals) to coexist in the same channel.

Generally, digital and analog modulations resort to basic modulation types:

- **Amplitude Modulation(AM)**, which changes the amplitude of the carrier
- **Frequency Modulation(FM)**, which changes the frequency of the carrier
- **Phase Modulation(PM)**, which changes the phase of the carrier

This kind of modulation is necessary to convey information to the receiver, assigning to each possible value of the information signal a different amplitude, frequency or phase.

Amplitude Modulation(AM)

The amplitude modulation is the simplest form of modulation.
The amplitude of an high-carrier signal(like a cosine signal) is varied according to the instantaneous amplitude of the modulating message signal $m(t)$.

Frequency Modulation(FM)

In frequency modulation, the frequency of the carrier signal is varied by the modulating signal $m(t)$, while the amplitude of the carrier signal is kept constant.
This means that the as the amplitude of the information signal varies, the carrier frequency varies as well. For example, if the amplitude of the information signal increases, the frequency of the carrier signal increases as well.

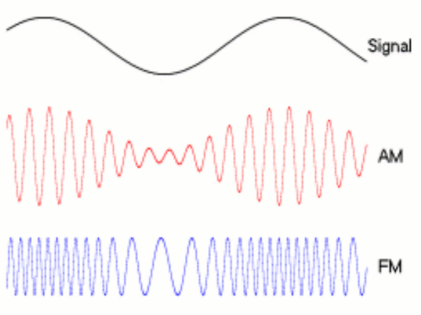


Figure 1.14: An example of a signal modulated in amplitude and frequency

Phase Modulation(PM)

Phase modulation is a form of modulation that encodes the signal $m(t)$ as a variation in the instantaneous phase of a carrier wave.
This means that the phase of a carrier is modulated to follow the changing in the signal amplitude of the message signal.

The peak amplitude and the frequency of the carrier signal are maintained constant, but as the amplitude of the message signal changes, the phase of the carrier changes correspondingly.

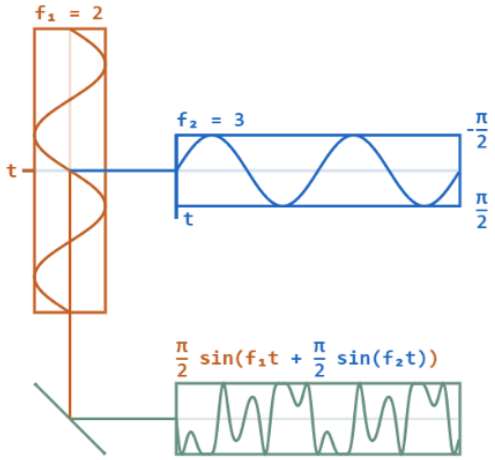


Figure 1.15: An example of a signal modulated in phase. The modulating wave(in blue) is modulating the phase of the carrier wave(in red), resulting in the PM signal(in green)

Analog-to-Digital modulations

Even if the world has turned to digital, transmitted signals are analog.
This means that the digital information has to be converted to an analog signal to be transmitted over the channel. But the receiver still need to understand the digital information from the received signal.
To be sure that the information can be recovered, the signal has to be modulated in a way that the receiver can understand the digital information. This can be done by varying some proprieties of the carrier signal, such as the amplitude, the frequency, or the phase, to represent the digital information.
The carrier signal is used to modulate the digital information, so we can distinguish between

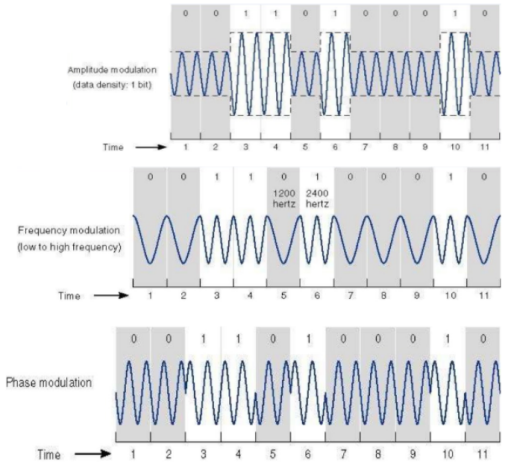


Figure 1.16: Some example of modulations used to represent digital information. From top to bottom: Amplitude Shift Keying, Frequency Shift Keying, Phase Shift Keying

different kinds of signals:

- the **the baseband signal**, which is the unmodulated signal, whose spectrum is centered around zero frequency
- the **passband signal**, which is the modulated signal, whose spectrum is centered around the carrier frequency

The baseband signal can be converted to a passband signal by multiplying it by a carrier signal with the desired frequency.

Baseband Signals

The simplest kind of digital modulation is the **Pulse Amplitude Modulation(PAM)**, which is a form of modulation where the message signal is encoded in the amplitude of a series of signal pulses.
For example, if we have a binary signal, we can encode the 0 as a low amplitude pulse $-A$, and the 1 as a high amplitude pulse A. The simplest pulse is a rectangular one, but other kind of pulses can be used.
If we have a binary PAM(2-PAM), the signal can be represented as:

- $s(t) = g(t) \rightarrow$ "1"
- $s(t) = -g(t) \rightarrow$ "0"

where $g(t)$ is the basic pulse shape.

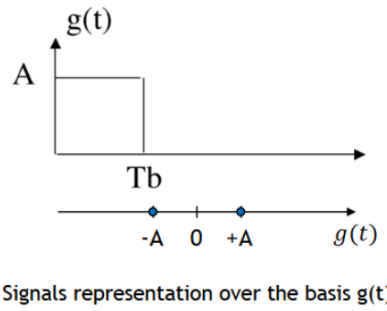


Figure 1.17: An example of a 2-PAM signal representation

M-ary PAM

WA 2-PAM signal can only represent 1 bit of information. To represent more bits, we can use M-ary PAM, where M is the number of different symbols that can be represented, while still using the same base signal.
For example, a 4-PAM signal can represent 2 bits of information by defining 4 levels of amplitude, and can be represented as:

- $s(t) = 3g(t) \rightarrow$ "00"
- $s(t) = g(t) \rightarrow$ "01"

- $s(t) = -g(t) \rightarrow$ "10"
- $s(t) = -3g(t) \rightarrow$ "11"

This definition can be generalized to:

$$s_i(t) = A_i g(t), \quad i = 1, 2, \ldots, M \tag{1.17}$$

allowing to represent $log_2(M)$ bits of information.

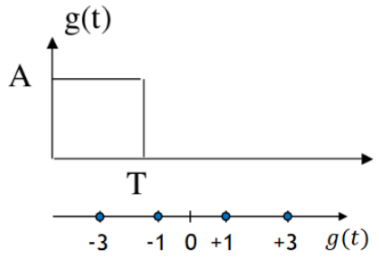


Figure 1.18: An example of a 4-PAM signal representation

Gray Coding

When using M-ary PAM, it is important to use a coding that minimizes the error probability. After all, Symbols that are close to each other in the signal space are more likely to be confused, so the choice of the number of symbols and the distance between them is important.

> **Gray coding** is a strategy to **mapping bits to symbols** that minimizes the probability of error.

Gray coding achieves 1-bit error correction, meaning that if a error is to occur, it will only affect 1 bit of the message with a high probability.

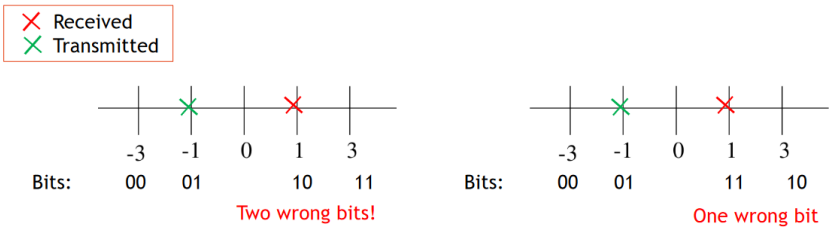


Figure 1.19: An example of a 4-PAM signal representation using Gray coding

Energy per bit

A measure of the energy efficiency of a modulation can be obtained by calculating the average energy per bit.
The energy per bit can be defined as

$$E_b = \frac{E_s}{log_2(M)} \tag{1.18}$$

where E_s is the energy of the signal, and M is the number of symbols, or in a more discursive way, the average energy per symbol divided by the number of bits carried by each symbol. This concept can also be visualized graphically, as the distance of a symbol from the origin in the signal space(as in figure 1.18), because it is proportional to energy of the symbol.

The energy per symbol can be calculated as

$$E_s = \int_0^T (S_m(t))^2 dt = (A_m)^2 \int_0^T (g(t))^2 dt = (A_m)^2 E_g \tag{1.19}$$

where $S_m(t)$ is the modulated signal, A_m is the amplitude of the modulated signal, $g(t)$ is the base pulse, and E_g is the energy of the base pulse.

For example, the average energy per symbol for the 4-PAM of figure 1.18 is

$$E_s = \frac{3^2T + 1^2T + 1^2T + 3^2T}{4} = 5T \tag{1.20}$$

Generally, the larger the energy, the larger the distance between the symbols, and the lower the probability of error(mistaking one symbol for another). On the other hand, the larger the number of symbols over the same bandwidth, the less energy is required to transmit each bit(because each symbol is closer and carries more bits).

Bandpass Signals

As previously stated, to transmit a baseband signal $s(t)$ trough a passband channel, we have to modulate it at a certain frequency f_c, by multiplying it by a sinusoidal carrier signal with that frequency, otherwise it will be centered around zero frequency.

Bandwidth Occupancy and efficiency

The shape of a signal determine the bandwidth it occupies.
For example, a rectangular function has a pulse spectrum that is a sinc function, shown in figure 1.20, which has a symbol rate of $1/T$(the amount of symbols per second).

Usually, a smaller base pulse will require less energy to transmit, but will occupy more bandwidth. A longer basic pulse will require more energy to transmit, but will occupy less bandwidth and have a lower bit rate.

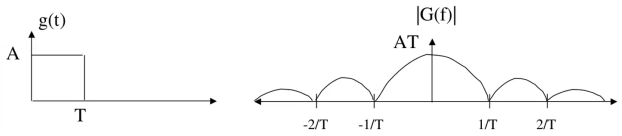


Figure 1.20: The spectrum of a rectangular pulse

With that in mind, ideally, we would like to choose a pulse shape $g(t)$ that minimizes the bandwidth occupancy, putting more energy in the lower frequencies, resulting in a smaller bandwidth.

For a pulse duration T, we can define the **symbol rate** as $R_s = 1/T$, and the **bit rate** as $R_b = R_s log_2(M)$(recall that $log_2(M)$ is the number of bits per symbol).
The bit rate efficiency can be defined as

$$\eta = \frac{R_b}{BW} = \frac{log_2(M)}{T} \times (\frac{T}{2}) = \frac{log_2(M)}{2} \text{bps/Hz} \tag{1.21}$$

where BW is the two sided bandwidth of the signal $BW = 2R_2 = \frac{2}{T}$.
As we can see, the bit rate efficiency is proportional to the number of bits per symbol ($\eta \propto M$) and inversely proportional to the pulse duration. The duration of the pulse is uninfluent.

We can also consider some examples:

- 2-PAM: $\eta = \frac{1}{2}$bps/Hz
- 4-PAM: $\eta = \frac{2}{2}$bps/Hz
- 8-PAM: $\eta = \frac{3}{2}$bps/Hz

As we can see, the bandwidth efficiency increases with the number of bits per symbol, because we are fitting more bits in the same bandwidth.
This also means that our signal will be more susceptible to noise, because the symbols are closer to each other, making it easier to mistake one for another. To reduce the probability of errors, we have to increase the energy per symbol, which will increase the bandwidth occupancy. For those reasons, there's a trade-off between bandwidth occupancy and energy efficiency.

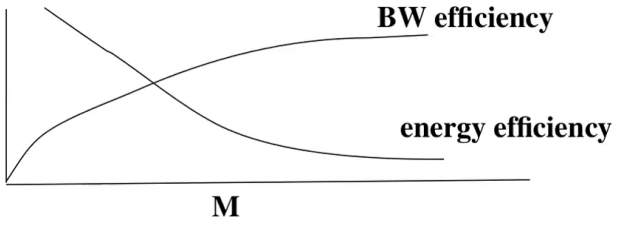


Figure 1.21: Trade-off between energy efficiency and bandwidth occupancy

Two-dimensional Modulation

As introduced in subsection 1.3.3, signals can be represented over two orthonormal basis. This means that we can represent a signal in a 2D plane, with the in-phase and quadrature components as the x and y axis.
This representation allows to represent the set of signals s_i(also called *constellation* over two orthonormal basis. A large constellation will allow to represent more bits per symbol, which results in a higher bit rate(bandwidth efficiency), but also a higher probability of error.

> The shape of the constellation is important, because it can be used to minimize the probability of error, by choosing a constellation that minimizes the distance between the symbols.

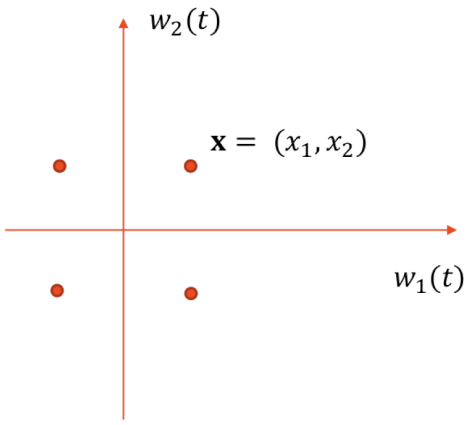


Figure 1.22: A constellation of 4-PAM signals

Furthermore, there are some common constellations that are used in practice, such as:

- **QAM**: Quadrature Amplitude Modulation, which is a PAM signal over two dimensions.
- **PSK**: Phase Shift Keying, which is a PAM signal over the phase of a signal, meaning that the amplitude is constant.

M-QAM

> M-QAM is a modulation that represents the signal as the sum of two signals, represented over two orthonormal basis. This means that we can have M symbols to modulate out signal.

Thus, we can represent $\sqrt{M}$ symbols over each axis, and the total number of symbols is M.
This means that a symbol, represented over two orthonormal basis, can be described as

$$S_m = (A_m^x, A_m^y), A_m^x, A_m^y \in \{+/-1, \ldots, +/-(\sqrt{M}-1)\}$$

for example:

- 4-QAM: $A_m^x, A_m^y \in \{+/-1\}$
- 16-QAM: $A_m^x, A_m^y \in \{+/-1, +/-3\}$

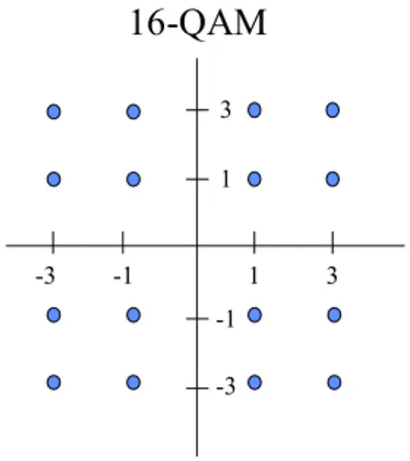


Figure 1.23: A 16-QAM constellation

We recall that with QAM, the amplitude and phase of the signal are modulated. This means that by using the same pulse shape $g(t)$, the bandwidth efficiency is the same of a M-PAM, because the number of bits per symbol is the same, but with a larger energy efficiency, thanks to the presence of another dimension.

By entering in the details, the in-phase and quadrature components of the signal are modulated by multiplying the components to orthogonal carriers, meaning that we can separate them easily afterwards. This operation also transform the signal to a bandpass one. This is accomplished by multiplying the A^x component by Cosine and the A^y component by Sine, and then summing the two signals.

The transmitted signal can thus be described as

$$U_m(t) = A_m^x g(t) cos(2\pi f_c t) + A_m^y g(t) sin(2\pi f_c t), m = 1, \ldots, M \tag{1.22}$$

or in a more discursive way, each component of the signal is the amplitude value over one axis multiplied by the base pulse(the basis), and then multiplied by a carrier signal at a given frequency.

M-QAM modulation and demodulation

The general schema of M-QAM modulation is shown in figure 1.24.
We receive in input a series of bits, which are associated with the corresponding symbols, one for each axis. The symbols used to modulate the pulse signal are then multiplied by the pulse shape, and then by the carrier signal. The two signals are then summed and

transmitted.

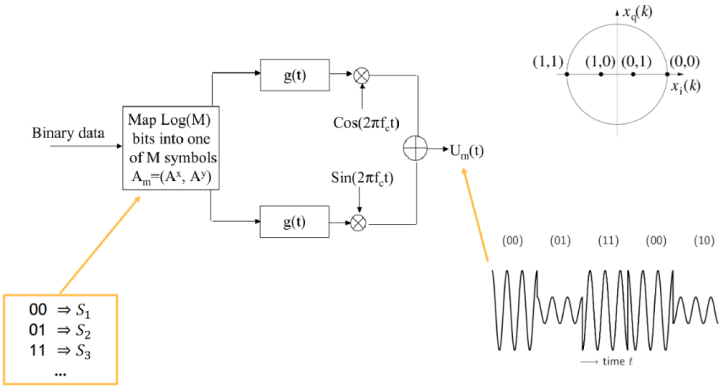


Figure 1.24: General schema of M-QAM modulation

As for what concerns the demodulation, we want to understand correctly the symbols that we receive, so we don't really care about the carrier signal. We can notice that the sin and cosine components of the signal are orthogonal, meaning that we are able to separate them easily: the cosine component disappears after filtering when multiplied by the sine component, and viceversa.

$$U_m(t) = A_m^x g(t) Cos(2\pi f_c t) + A_m^y g(t) Sin(2\pi f_c t),\ m = 1..M$$

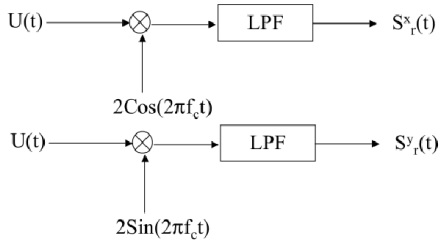


Figure 1.25: General schema of M-QAM demodulation

Phase Shift Keying

> **Phase Shift Keying**(PSK) is a PAM signal over the phase of a signal.
> In a PSK constellation, the amplitude of the signal is constant, meaning that all the signals have the same energy.

This means that the symbols are equally spaced across a circle of radius $\sqrt{E_S}$, where E_S is the energy of the signal.
Symbols are thus equally spaced to minimize the probability of error.

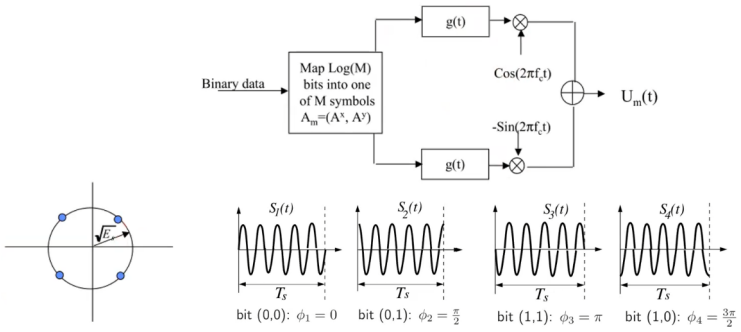


Figure 1.26: General schema of M-PSK modulation

The modulation process is similar to the one of M-QAM. We can notice that, because all the amplitude values are so similar, resulting the waveforms have the same energy.

Power amplifiers

After modulation, the signal is usually amplified by a **power amplifier** to increase the power of the signal.

Because we cannot have infinite energy, we can only amplify the signal to a certain level,

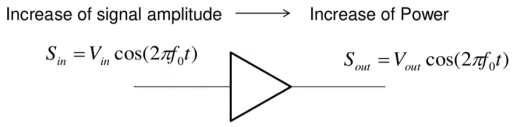
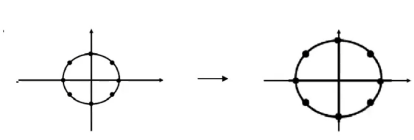


Figure 1.27: General schema of a power amplifier

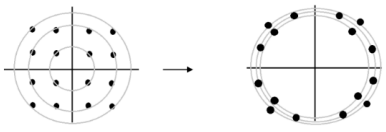
which is called **saturation level**.

This has some effect on the constellations of a signals:

- when working with **PSK** signals, the constellation in amplified uniformly even when working in saturation.
- when working with **QAM** signals, the constellation is scaled differently, making the probability of error higher.



(a) PSK constellation after amplification



(b) QAM constellation after amplification

Figure 1.28: Effect of power amplification on a signal

When working under the saturation level, the power amplifier is linear, meaning that the output signal is proportional to the input signal, and no distortion occurs, but the amplifier is underexploited.

Thus, there is a trade-off between transmitted power and signal quality.

1.4.2 AWGN channel and equalization

Recall that to transmit information, we need to send a signal over a channel, in our case a wireless one.
We already briefly discussed the disturbance of a signals in subsection 1.2.2, but the main challenges can be summarized as:

- share the medium via **multiplexing**
- fight **noise** and **channel impairments**

The major sources of errors in the channel are essentualy two:

- **Termal noise**(AWGN), which is the result of the thermal agitation of the electrons in the receiver. It disturbs the signal in an additive fashion and occupies all the frequency band. Is also modelled by a Gaussian random process.
- **Inter-Symbol Interference**(ISI), which is the result of the signal being reflected by obstacles, and thus arriving at the receiver with a different phase and amplitude.

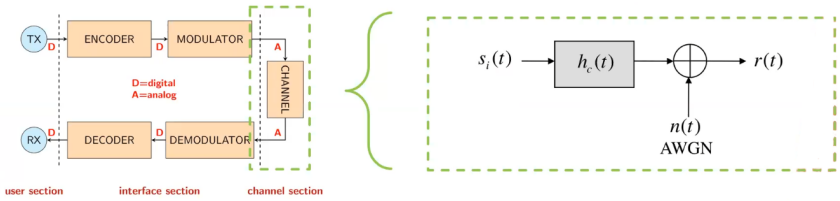
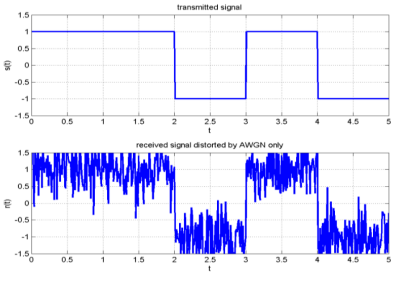
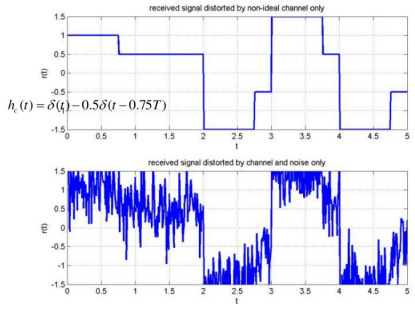


Figure 1.29: Effects of the channel on a signal, $h(t)$ is a filter that models the channel with it impusle response, and $n(t)$ is the noise

We can say that the channel ruin the signal. This is evident when looking at the spectrum of the signal, which is spread by the channel, shown in figure 1.30.



(a) Effect of the noise on the signal



(b) Signal spectrum distorted by inter-symbol interference, modelled by $h_c(t)$

Figure 1.30: Effect of the channel on the spectrum of a signal

Receiver Task

We want to design a receiver that can mitigate (or even revert) those two effects, to be able to correctly understand the signal.
To do so, we have to **demodulate** the signal and **equalize** it.
This process can be carried out in 3 steps:

1. Improve the signal-to-noise ratio(SNR) using a **matched filter**
2. Reduce Inter-Symbol Interference(ISI) using an **equalizer**
3. Sample the recovered waveform and guess the transmitted symbol

After those steps, we still have an imperfect signal wave, so we need to **detect** the transmitted symbol, thresholding the signals to remove outliers introduced in the channel and decide with symbol was transmitted.

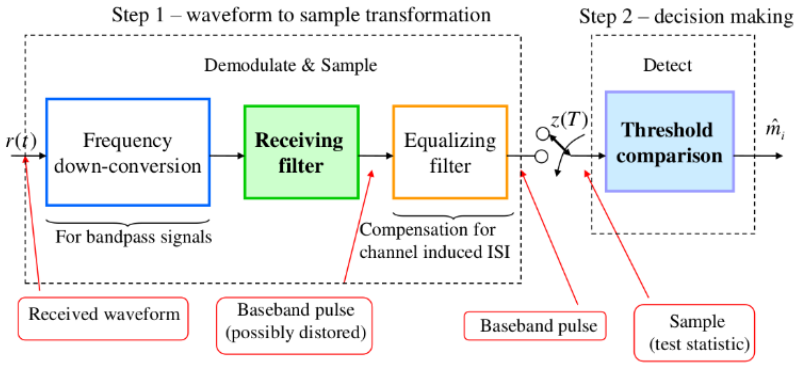


Figure 1.31: General steps that a receiver has to carry out

Maximize SNR

> To maximize the signal-to-noise ratio, we can use a **matched filter** $h(t) = g(T-t)$

Turns out that the input response of the optimal filter is the time-reversed version of the pulse shape.
The receiver must know the basic pulse shape to be able to use the matched filter.

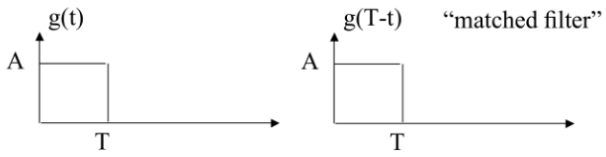


Figure 1.32: Effect of the matched filter on the signal. They are kind of the same because the function is symmetrical

Minimize ISI

Inter-Symbol Interference is the result of the filtering effect of the channel, and can be defined as

$$H_c(f) = |H_c(f)|e^{j\phi_c(f)} \tag{1.23}$$

where $|H_c(f)|$ is the amplitude response of the channel, which is non-constant meaning that it distorts the amplitude of the signal, and $\phi_c(f)$ is the phase response of the channel, which is non-linear, meaning that it distorts the phase of the signal.

To revert those effects, we can use an **equalizer** to compensate for the distortion of the channel, which ideally would be the inverse of the channel($H_e(f) = \frac{1}{H_c(f)}$).
Applying it to the signal allows us to get an approximation of the original symbol that was transmitted $\hat{S}_i(t)$.

To build the equalizer we still need to know the frequency response of the channel. To do that, we can try to estimate the channel based on known characteristics of the signal. Usually we don't know them, so we introduce **pilot symbols** in the signal, which are known signals to the receiver, and can be used to estimate the channel.

Fading

Depending on the channel, its impulse response can vary very slowly or very quickly.
When the impulse response varies slowly, we talk about **slow fading**, and when it varies quickly, we talk about **fast fading**.
Depending on this, pilots symbols may need to be transmitted more or less frequently.

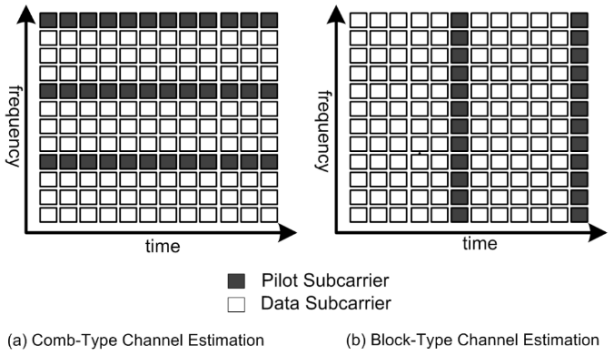


Figure 1.33: Pilot transmission scheme

1.4.3 Received symbols and decision regions

After equalizing the signal, we would like to decide which symbol was transmitted.
As previously stated, we get a set of possible symbols $S_m \in \{S_1, S_2, ..., S_M\}$, and we want to decide which one was transmitted, adopting a statistical approach.

Symbol Detection

Mathematically speaking, we have an hypothesis testing problem, which wants to minimize the probability of decision error, meaning that we want to choose the symbol that maximizes the probability of the received signal given the transmitted symbol.

This is known as the **Maximum a posteriori probability** (MAP) decision rule, which simplify that rule. In fact, turns out that maximizing the probability of the received signal given the transmitted symbol is equivalent to minimizing the distance between the received signal and the transmitted symbol, under certain conditions.
This is also known as **Minimum distance decoding** $d_{rS_m} = (r - S_m)^2$, where r is the received signal(actual value of the signal plus the noise).

Take for example the case of a 2-PAM signal, shown in figure 1.34. The noise alter the received signal, so we can define it as $r = S_m + n$, where n is the noise. We also know that the noise is a Gaussian random process, so we can suppose that the noise is more likely will be zero, and its less likely will be far from it.
This means that it is possible to assume that the received signal r is more likely to be the symbol S_m that is actually closer to it.
So, in this case, we can define the decision region as the interval $[-\infty, 0)$ and $(0, \infty]$, and the decision rule as

$$\hat{S_m} = \begin{cases} S_1 & \text{if } r < 0 \\ S_2 & \text{if } r \geq 0 \end{cases} \tag{1.24}$$

This can also be generalized to the case of M-PAM signals, because the general idea is still valid.

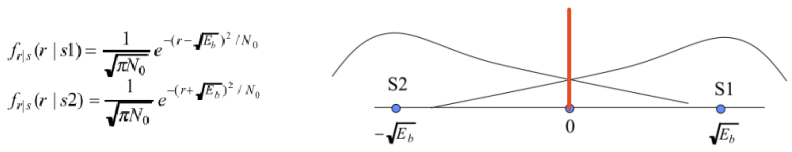


Figure 1.34: Example of signal detection

Probability of error

The Minimum Distance Decoding rule adopts a statistical approach to decide which symbol was transmitted, so it is possible to make errors.
In general, the probability of error P_e between two symbols separated by a distance d is

$$P_e = Q\left(\frac{d}{2\sigma}\right) \tag{1.25}$$

where Q is a complex function, and N_0 is the noise density.
The probability of error can be minimized by increasing the distance between the symbols.

Based on that we can compute the probability of error per bit, or Bit Error Rate(BER), for each modulation scheme.

Under certain condition, for example grey coding, the BER can be approximated as

$$BER = \frac{P_e}{\log_2(M)} \tag{1.26}$$

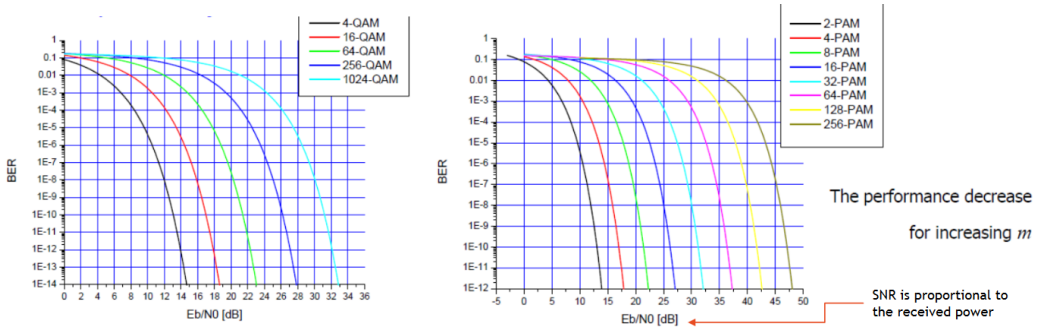


Figure 1.35: Examples of BER plots for different modulation schemes. The BER is higher at the same SNR for denser modulation schemes

1.4.4 Signal Attenuation and Link Budget

During the propagation of a signal, especially in wireless medium, it suffers from an attenuation L, which is the loss of power of the signal.
In general, it depends on many environmental factors, such as the distance between the transmitter and the receiver, the frequency of the signal, the presence of obstacles, and so on.

The relationship between transmission and reception power is given by the $P_R = P_T/L$, where P_R is the received power, P_T is the transmitted power, and L is the attenuation.
Given this relationship, we can define the Signal-to-Noise Ratio(SNR) perceived by the receiver as

$$SNR = \frac{E_b}{N_0} \tag{1.27}$$

where E_b the ratio between the Received Power and the Bit Rate, and N_0 is the noise density.
In general, to deal with this problem, we use antennas and amplifiers to increase the power of the signal and compensate for the attenuation. The shape of the antenna allows to modulate the rate of the signal: for example a wider antenna allows to spread the signal in a isotropic way and reach a wider area, while a more focused antenna allows to reach a more specific one.
This is measure by the **beamwidth** of the antenna θ_B, which is a measure of the directivity of the antenna.

The smaller the beamwidth, the more focused the antenna is, hence we yield a higher gain, but we also have a smaller area of coverage.
The larger the beamwidth, the more isotropic the antenna is, hence we yield a lower gain, but we also have a larger area of coverage.

For example, a parabolic antenna has a very small beamwidth, so $\theta_B \approx 70\gamma/D$ where D is the diameter of the antenna.
In general, the gain G_T is proportional to the area of the antenna, so $G_T \propto 1/D^2$. Doubling the diameter of the antenna, we get a 4 times increase in the gain.

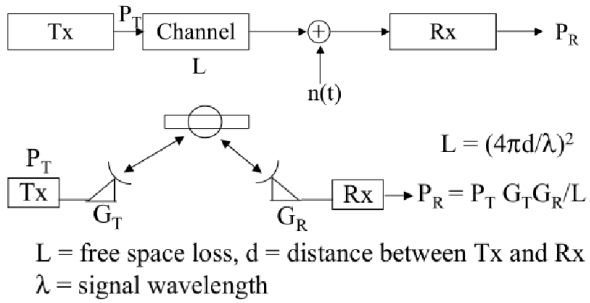


Figure 1.36: Example of signal attenuation over a wireless medium

1.4.5 Multiple Access Schemes

As previously discussed in section 1.3.9, we to trasmitt multiple signals over the same same shared medium, we have to use a method called **multiplexing**.

Multiplexing allows to divide the available bandwidth over multiple logical channels. The multiplexed signals are then transmitted over the same medium, and then demultiplexed at the receiver.

Multiplexing on itself is not enough to allow multiple users to transmit over the same medium. We also need to use a method called **Multiple Access**, which allows multiple users to transmit over the same medium.

Multiplexing deals with **combining signals**, while **multiple access** deals with allowing **multiple users** to access and share a communication medium.

Frequency Division Multiplexing/Multiple Access

As previously discussed, FDM allows to divide the available bandwidth over multiple logical channels.
In the case of multiple access, we can use FDM to divide the available bandwidth over

multiple users, and then modulating each user's signal over a different carrier frequency.
This is done by assigning a different frequency band to each user, so that they can transmit over the same medium without interfering with each other.
This is the case of **Frequency Division Multiple Access** (FDM/FDMA).
With this schema, each user get access to the full bandwidth for the whole time, being able to completely avoid interference, but if the user is not transmitting, the channel is wasted.
Furthermore, it is necessary to have a guard band between the channels to avoid interference.
All wireless systems use this scheme.

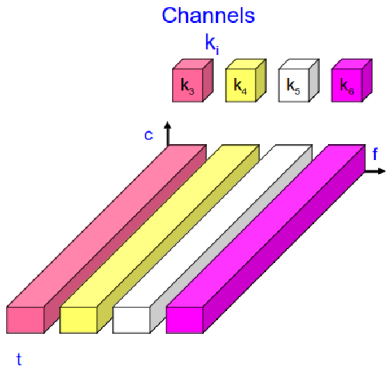


Figure 1.37: Example of FDMA

Time Division Multiplexing/Multiple Access

TDM is a method that allows to divide the available bandwidth over multiple logical channels, by assigning a different time slot to each user.
This slot is allocated to the user even if it has no data to transmitt. For this method to work correctly some kind of synchronization is needed, but each communication channel has access to the full bandwidth, even if it is only for a fraction of the time.
In syncronous TDMA, many slots are wasted. There is a **statiscal version** of TDMA,

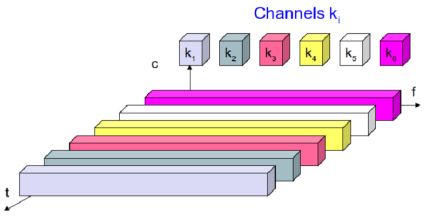


Figure 1.38: Example of TDMA

called **STDM**, which reassigns the slots to the users that need them.
This is done by scanning the input lines an collecting data until the frame is full, and then transmitting the frame.
This method allows to avoid wasting slots, but it is more complex to implement, requiring scheduling algorithms.

Code Division Multiple Access

CDMA is a method that allows to divide the available bandwidth over multiple logical channels, by assigning a different code to each user.
This method allows to exploit orthogonality between signals(allowing to separate them). Each channel has an unique code, while sharing the same spectrum(all of it) at the same time.
Each channel is assigned a different code, and the receiver uses the unique binary code c_i of a sender to separate the signal from the others.

For example a multiplexed signal

$$s_{mux}(t) = s_1(t)c_1 + s_2(t)c_2 + s_3(t)c_3$$

is demultiplexed by the receiver using the code c_1 of the first sender

$$\langle s_{mux}(t), c_1 \rangle = s_1(t)$$

In this way, we are able to achieve a great bandwidth efficiency, with no need for coordination or synchronization while also getting a good degree of protection against interference.
There are also some drawbacks: we have lower user data rates, and the system is more complex because the signals are more complex to regenerate.

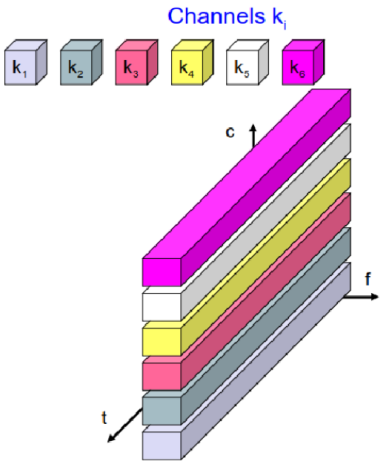


Figure 1.39: Example of CDMA

TDM/A + FDM/A

It is also possible to combine the two methods. We can combine the time and frequency division multiplexing, allowing to divide the available bandwidth over multiple logical channels, by assigning a different time slot and frequency band to each user.
With this solution, we are able to achieve a better degree of protection against tapping and frequency selective interference, at the cost of higher data rates(compared to CDMA).

This method also requires a precise coordination between the users.

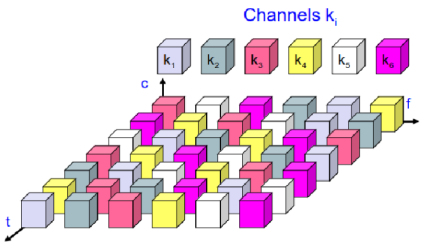


Figure 1.40: Example of TDMA/FDMA

1.5 Source and channel coding

A brief overview: we have a source that generates a message, which is then encoded by the source encoder, to limit the amount of bits transmitted and the effects of noise.
Those bits are then transformed into waveforms, which are altered by the channel, and then transformed back into bits by the channel decoder, after inverting the effects of the channel.

Part of this operation are carried out in two steps:

- **Encoding**: which aims to produce a compressed representation Y of the original message X. Usually encoding is simply a function f that maps X to Y.
- **Decoding**: which aims to recover the original message X from the representation Y. Usually decoding is simply a function f^{-1} that maps Y to X.

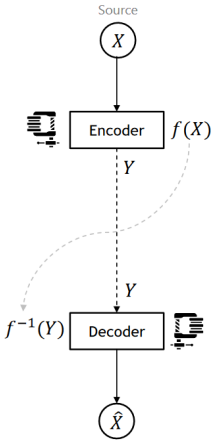


Figure 1.41: General schema of encoding and decoding

1.5.1 Source Coding

We would like to choose a encoding function f that can be reverted while also **reducing redundancy** in the message.

Source coding, also known as data compression, aims to represent information or data in a more compact form to reduce redundancy and save storage space or transmission bandwidth.

This is possible because many real-world datasets exhibit patterns or repetitions that can be efficiently encoded

Symbol	Frequency	Variable length Code	Fixed length Code
A	6	0	00
B	3	10	01
C	2	110	10
D	1	111	11

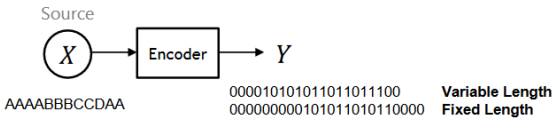


Figure 1.42: Example of source coding

Figure 1.42 shows an example of source coding. The example supposes that a sequence of letters is being transmitted. A possible uncompressed representation of the letters would be assigning a two bit code to each letter(fixed length code).
An alternative strategy exploits the fact that some letters are more frequent than others, assigning shorter codes to the most frequent letters(variable length code). With this solution, we are able to reduce the number of bits needed to transmit the message.

Compression Types

There are two main compression classes:

- **Lossless Compression**: the original data can be perfectly reconstructed from the compressed data($\hat{X} = X$). This is useful when the original data must be preserved exactly, such as in text or executable files. Some examples of lossless compression includes ZIP, RAR, and PNG.
- **Lossy Compression**: the original data cannot be perfectly reconstructed from the compressed data($\hat{X} \approx X$). This is useful when the original data can be approximated or when some loss of quality is acceptable, such as in images or audio files.

As there are no officials compression formats, the specifications are defined by the fidelity requirements of the application. Generally, there is a trade-off between the compression efficiency and the computational complexity of the algorithm.

1.5.2 Error Detection and Correction

After compression, all the redundancy of the message has been possibly removed, and the bits that make up the message are the ones to be transmitted.
Ideally, the message should be received unaltered, but we know that the channel can introduce errors in the message.

Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data. But those schemes add some redundancy to the message.
For doing this, there are two main approaches:

- **Automatic Repeat request(ARQ)**: it implements acknowledgements and timeouts to ensure that the message is correctly received.
- **Forward Error Correction(FEC)/Channel coding**: it adds redundancy to the message to allow the receiver to correct errors without the need for retransmission.

Channel Coding

Channel coding aims to detect and correct errors that may occur during transmission.
It adds redundant bits to the original message, creating a coded message that contains extra information for error detection and correction.
The effectiveness of a channel code is often measured by its error correction capability, indicating the maximum number of errors that can be corrected within a codeword.

There are two main types of channel coding:

- **Block Codes**: the message is divided into blocks of fixed length, and a fixed number of redundant bits are added to each block. The receiver uses the redundant bits to detect and correct errors.
- **Convolutional Codes**: the message is encoded using a convolutional encoder, which adds redundant bits based on the current and previous bits of the message. The receiver uses a Viterbi decoder to detect and correct errors.

Block Codes

A block code acts on block of k bits of input data to produce n bits of output data. This operation is carried out by the channel encoder.
It simply adds $n - k$ bits of redundancy to the original message, which can be used by the receiver to detect and correct errors.

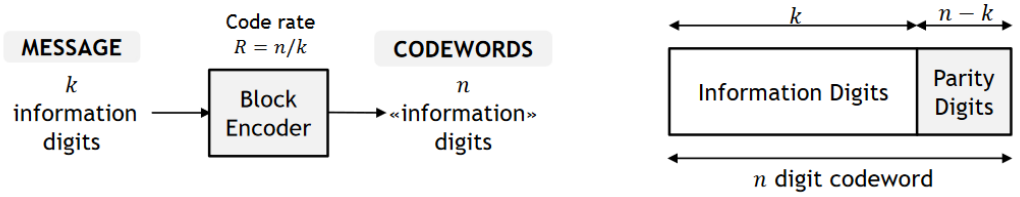


Figure 1.43: Example of block code

Chapter 2

Security at the physical layer

Part II

Hardware