LABORATORY TEST

BASIC SIMULATIONS OF DIGITAL COMMUNICATION SYSTEMS

Objectives

Information transmission in digital communication systems, both baseband and bandpass, can be affected by several kinds of distortion. Some sources of distortion, such as noise or linear distortion due to the channel, are inherent to the transmission. On the contrary, others might be a consequence of systematic errors in the communications equipment.

Along this first exercise, which will last for two sessions, some of those effects will be simulated in Matlab. In this way, students will learn the fundamental notions required to simulate a digital communications system at a basic level and to analyze the impact of some of its parameters on the overall performance of the system through those simulations.

1 Matlab files

Matlab files which allow the generation of several signals of interest are provided as an aid to carry out the tasks requested in this practice. A brief summary of the files and their main purpose can be found in the following table.

Name	Description
generate_bits	Generates a random sequence of bits with the specified length
generate_pam	Transforms a sequence of bits into symbols of a M-PAM constellation for a given value of M
generate_qam	Transforms a sequence of bits into symbols of a M-QAM constellation for a given value of M
detector_pam	Implements the decisor for M-PAM constellations for a given value of M
detector_qam	Implements the decisor for M-QAM constellations for a given value of M
generate_noise	Generates a discrete sequence of Gaussian noise, either baseband of passband, with the specified length and value of N_0
gray_pam	Transforms a sequence of bits into symbols of a M-PAM constellation using Gray encoding for a given value of M
gray_qam	Transforms a sequence of bits into symbols of a M-QAM constellation using Gray encoding for a given value of M
pam_gray	Transforms a sequence of M-PAM symbols to bits using Gray encoding for a given value of M

qam_gray	Transforms a sequence of M-QAM symbols to bits using Gray encoding for a given value of M
sequential_pam	Transforms a sequence of bits into symbols of a M-PAM constellation using sequential binary encoding for a given value of M
sequential_qam	Transforms a sequence of bits into symbols of a M-QAM constellation using sequential binary encoding for a given value of M
pam_sequential	Transforms a sequence of M-PAM symbols to bits using sequential binary encoding for a given value of M
qam_sequential	Transforms a sequence of M-QAM symbols to bits using sequential binary encoding for a given value of M
generate_observation_Viterbi_pam	Generates the sequence of observations to be used within the Viterbi algorithm for ML detection, for a given value of M and discrete equivalent channel

To find a more detailed description of each of those functions, the help command in Matlab can be employed.

1.1 Other Matlab functions of interest for this practice

Some built-in Matlab functions can be also useful to solve some of the points proposed along this practice. Some of them are:

Name	Description
rand	Generate random numbers according to a uniform distribution
randn	Generate random numbers according to a Gaussian distribution
plot	Represent a continuous-time signal in time
digplot	Represent a discrete-time signal in time
semilogy	Represent a function with log-scale in the Y-axis
find	Returns the indexes of non-zero elements within a vector
conv	Computes the convolution of two discrete sequences
print	Allows saving a figure in several formats
inv	Computes the inverse of a matrix. Usage of function inv is not recommended for chained operations though. Employ A\ B and B/A instead of inv(A)*B and B*inv(A) respectively
pinv	Computes the Moore-Penrose pseudoinverse of a matrix
transpose	Obtain the transpose of a matrix. Can be written also as A.'
ctranspose	Obtain the conjugate-transpose of a matrix. Can be written also as A'

Again, to find a more detailed description of each of those functions, the help command in Matlab can be employed.

2 Effect of noise - E_b/N_0 ratio

Noise is one of the main factors which cause a degradation in the performance of a communications system. The amount of degradation will depend on the particular statistical distribution of the noise process, as well as on its power relative to the power of the transmitted signal. The noise process $n(t)$ is commonly modeled as a white, Gaussian stochastic process with power spectral density $N_0/2$, which is added to the modulated signal. If the receiver uses a filter $f(t)$ whose deterministic autocorrelation function $r_f(t) = f(t) * f(-t)$ fulfills the Nyquist criterion with symbol period T, then the noise process sampled at the output of the demodulator, $z[n]$, is also Gaussian and white. Thus, its power spectral density, $S_z\left(e^{j\omega}\right) = \sigma_z^2$, is constant in the entire spectrum. Moreover, σ_z^2 will take value $\sigma_z^2 = \frac{N_0}{2}$ for baseband transmissions and $\sigma_z^2 = N_0$ for passband transmissions[1].

There exist several ways to measure the relation between the transmitted signal power and the noise power. One of the most commonly used is the so called *bit energy / N_0 ratio*, or E_b/N_0 ratio. It is defined as the quotient between the average energy transmitted per bit and the value of N_0. Therefore:

$$\frac{E_b}{N_0} = \frac{E_s}{m \cdot N_0}$$

Where $E_s = E[|A[n]|^2]$ is the average energy per symbol in the constellation being transmitted. Note that when such constellation contains M different symbols, then $m = log_2(M)$ is the number of bits per symbol so that $E_b = E_s/m$. The E_b/N_0 ratio is usually expressed in decibels:

$$\frac{E_b}{N_0}(dB) = 10\log_{10}\left(\frac{E_b}{N_0}\right)$$

As the noise power increases, the average distance between the transmitted symbol, $A[n]$, and the observation obtained at the output of the demodulator, $q[n]$, increases as well. This in turn causes an increment in the error probability. One way to check the amount of noise present in passband digital communication systems is by using a scatter plot; a graph in which the in-phase component of the observation, $q_I[n]$, is represented in the X-axis, whereas the in-quadrature component, $q_Q[n]$, is shown in the Y-axis.

In this section, students are asked to generate a scatter plot for a 16-QAM constellation sent over a Gaussian channel without linear distortion, using three distinct values of the E_b/N_0 ratio:

- 30 dB
- 10 dB
- 3 dB

Note that for each value of E_b/N_0, N_0 must be computed before being able to obtain the scatter plot.

[1] Recall that in this case the noise process is complex, with the real part modeling the noise in the in-phase component, $z_I[n]$, and the imaginary part accounting for the noise in the in-quadrature component, $z_Q[n]$. Furthermore, both noise components are independent and with variance $\sigma_{z_I}^2 = \sigma_{z_Q}^2 = \frac{N_0}{2}$. In other words, the sampled noise is a circularly symmetric complex stochastic process.

3 Effects of Inter-Symbol Interference

One of the most common phenomena affecting digital communication systems is *Inter-Symbol Interference* (ISI), which is due to the linear distortion introduced by the channel. The main consequence of the presence of ISI in digital communication systems is that the constellation in the receiver is different from the original constellation in such a way that some "fake" symbols are artificially "created" by the ISI. This can be also observed easily by using a scatter plot.

The amount of degradation caused by the ISI in a digital communications system depends on several factors such as the length of the discrete equivalent channel, the relative amplitude of each delay with respect to the main tap and the order of the constellation being used.

In this section, the impact of those factors will be studied by generating scatter plots obtained using several distinct discrete equivalent channels.

Firstly, students are asked to obtain a scatter plot for a transmission with $E_b/N_0 = 30$ dB and a 4-QAM constellation using the following discrete equivalent channel with length 2 (memory of the channel is therefore $K_p = 1$):

$$p[n] = \delta[n] + a \cdot \delta[n-1]$$

Three different values of a must be employed: $a \in \{\frac{1}{16}, \frac{1}{8}, \frac{1}{4}\}$.

Discuss the effect of increasing the value of a in the constellation observed by the receiver and explain how will that affect the performance of the communications system.

Afterwards, students shall repeat the previous question but using a 16-QAM constellation instead. Discuss the differences with respect to the case when a 4-QAM constellation was employed and explain what can be concluded from those changes.

Finally, the impact of increasing the length of the discrete equivalent channel will be explored. For this, the two previous questions will be repeated but using the following discrete equivalent channel:

$$p[n] = \delta[n] + a \cdot \delta[n-1] + \frac{a}{2} \cdot \delta[n-2]$$

Where the values of a are the same as before.

4 Impact of noise on the error probability

Noise does not affect different constellations in the same way since, for a fixed value of E_b/N_0, the relative distance between symbols (or the noise level if that distance is kept constant) changes. Besides, at bit level, the particular mapping of bits to symbols significantly influences the system. Those effects will be studied along this section. For this, students will obtain the *Bit Error Rate* (BER) as a function of E_b/N_0 in dB for several distinct constellations using both Gray encoding and binary sequential encoding to map bits to symbols in the constellation[2].

[2] To obtain a reliable estimation of the error probability at least 100 errors must be observed throughout the simulation. Otherwise, such simulation cannot be considered to be statistically significant. To avoid having to

Firstly, students are asked to obtain the BER for values of E_b/N_0 within 0 and 20 dB in steps of 2.5 dB, using Gray encoding, for the following constellations:

- Baseband 2-PAM, 4-PAM, 8-PAM, and 16-PAM (all four must be plotted in the same figure).
- Passband 4-QAM, 16-QAM, and 64-QAM (all three must be plotted in the same figure).

Recall from the theoretical lectures that the error probability decays exponentially as E_b/N_0 is increased. As a consequence, it is strongly recommended to obtain those plots using a logarithmically-scaled Y-axis. For this, you may use the Matlab command **semilogy**.

It is also interesting to observe how the relation between the symbol error probability, P_e, and the bit error probability, BER, evolves as E_b/N_0 increases. For this, students are asked to plot the aforementioned relation only for a 16-QAM constellation. Discuss the results and extract the appropiate conclusions.

Finally, students shall repeat the previous questions using binary sequential encoding instead. Comment the differences and discuss the cause of those changes.

5 Error probability in non-coherent receivers

A receiver is said to be coherent when the carriers used during the demodulation process in the receiver are in-phase with the carrier used by the transmitter in the modulation process. Whenever there is a certain phase difference, we say that such receiver is non-coherent. The effect which non-coherent reception has on the system is to rotate the original constellation by an angle equal to the phase difference between the carriers used by the receiver and those used by the transmitter. Of course, this rotation implies that the overall performance of the system will degrade.

In order to observe this effect, students are asked to represent the BER as a function of the phase difference between carriers within a range from 0^o to 90^o in steps of 5^o. Employ a 16-QAM constellation with Gray encoding and $E_b/N_0 = 12$ dB.

6 Impact of ISI on the error probability

The presence of ISI has a strong effect on the error probability of digital communication systems.

Students are requested to obtain plots of the BER for values of E_b/N_0 within 0 and 20 dB in steps of 2.5 dB, using Gray encoding, for a 16-QAM constellation transmitted over a discrete equivalent channel defined by the following expression:

$$p[n] = \delta[n] + a \cdot \delta[n-1]$$

With a taking the values $a \in \{\frac{1}{16}, \frac{1}{8}, \frac{1}{4}\}$. Compare the results with those obtained without ISI, making any comments you consider relevant.

simulate the transmission of too many symbols to meet such requirement, students do not need to consider error probabilities below 10^{-6}.

7 Maximum Likelihood sequence detection - Viterbi algorithm [Optional]

Whenever there is ISI, the optimal implementation of the receiver can be shown to be based on detection of entire sequences under the Maximum Likelihood (ML) criterion, which can be implemented using the Viterbi algorithm. During this section, students will implement a ML sequence detector in Matlab and study the performance it can achive.

To be precise, students shall implement such detector for the following discrete equivalent channel:

$$p[n] = 0.8\delta[n] + 0.2\delta[n-1]$$

When the constellation to be used is 4-PAM. The sequence of transmitted symbols is to be split in pieces of length of $L = 10$ samples and the cyclic header sent to set the initial state in each block will be the symbol +3. Using this particular receiver, students are asked to obtain the BER as a function of E_b/N_0 for values between 0 and 20 dB.

8 Channel equalization [Optional]

In many practical cases, the complexity of including a ML sequence detector in a particular receiver may be too high. In those cases, one particular approach is to use a channel equalizer, a sub-optimal solution but much easier to implement. In this final section, the operation of a digital communications system with channel equalization will be studied.

For the discrete equivalent channel discussed in the previous section, students are asked to design channel equalizers with 4 coefficients ($K_w = 3$), and a delay $d = 2$ for the following distinct criteria:

- Zero-Forcing criterion (ZF)
- Minimum Mean-Square-Error (MMSE)[3]

For the previous equalizers, students are requested to evaluate the performance using a 4-PAM constellation with E_b/N_0 taking values in the range from 0 to 20 dB. Discuss the results and compare them with those obtained without ISI and with ISI being dealt with ML sequence detection. Also, observe how do the coefficients of the MMSE equalizer evolve with E_b/N_0 and compare them with those of the ZF equalizer, which is the same for all values of E_b/N_0. Do you observe any particular phenomena? Why does that happen?

References

- *Comunicaciones Digitales.* A. Artés, F. Pérez González, J. Cid Sueiro, R. López Valcarce, C. Mosquera Nartallo y Fernando Pérez Cruz. Ed. Pearson Educación. 2007.

[3] Note that, in this case, the equalizer depends on E_b/N_0 so each time that ratio changes the equalizer has to be recomputed.

- *Communication Systems Engineering.* J.G. Proakis y M. Salehi. Prentice-Hall. 1994.