École Polytechnique de Louvain

LELEC2880
Modem design

Coded OFDM transmission on a frequency selective channel

Group M

Jose Antonio Simón Rodríguez

Adrián Pérez Expósito

May 17, 2020

Index

Introduction

During the last years, mobile communications have shown a strong growth, giving way to more robust new techniques with better performance. A clear example of this is OFDM (Orthogonal Frequency Division Multiplexing). Since its very beginnings, this modulation scheme has been the object of several studies which have led to one of the most used techniques in mobile communications.

At the present, the massive demand of wireless communication services has led to the need to have systems that support high data rate. In the analysis of parameters involved with the data rate, numerous important investigations have been developed. The transmission power and the electromagnetic spectrum are resources than in wireless environment are of important consideration.

OFDM (Orthogonal Frequency Division Multiplexing) is the modulation of choice for most nowadays systems. It makes equalization very easy with the use of the cyclic prefix, and converts the transmission through a wideband frequency selective channel into multiple parallel and independent frequency at subchannels. This offers a lot of flexibility for per-subcarrier processing, allowing for instance bit and power allocation, coding across the frequencies to obtain diversity, or simplified MIMO schemes.

For all the steps of this project, we consider a common OFDM transmission with the following characteristics.

- The number of subcarriers is equal to N = 128.
- The subcarriers are assumed to be modulated around a carrier frequency of 2 GHz.
- The subcarrier spacing is 15 kHz.
- The cyclic prefix is fixed at L = 16 samples. The assumptions on the channel will be different for each part but in all cases, the cyclic prefix is assumed to be long enough (and the frame synchronization sufficiently accurate) so that the transmission is performed without inter-carrier and inter-symbol interference.

Objectives

The objective of this project is to study several parts of an OFDM system. The study is based on mathematical derivations, computer calculations and simulations (using MATLAB for instance). It focuses on the following points:

- Implementation of an OFDM chain on a frequency selective channel.
- Adaptive modulation with bit and power allocation.
- Channel estimation.
- Coding across the frequencies and optimal decoding.

Step 1 Basic OFDM chain.

In this step it has simulated a basic OFDM chain on an ideal AWGN channel, it has generated 300 symbols OFDM and it has passed these symbols for OFDM communication chain. Then, it has computed the bit error rate (BER) and it has computed this for different values of SNR (Signal to Noise Ratio) and finally it has compared our graph of BER VS SNR between the theorical BER.

It has used this model of basic OFDM chain for all the step.

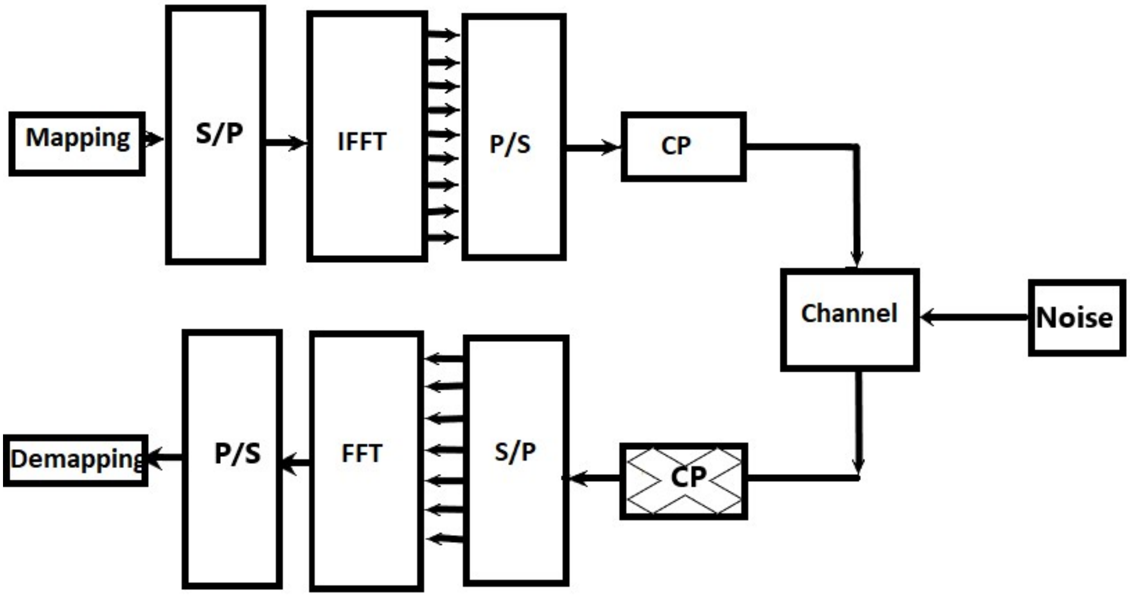

Figure (1): Model of basic OFDM chain

First, it has generated the initial variables that it has used in this step.

Then, it has generated a matrix of random binary data between 0 and M-1, after this, it did a 4-QAM modulations remember that it has 128 subcarriers, and each symbols OFDM have 128 bits modulated in 4-QAM constellations.

Here, it can see the 4-QAM modulation for mapping the symbols OFDM, this is 4 symbols are equally spaced in quadrature and in phase.

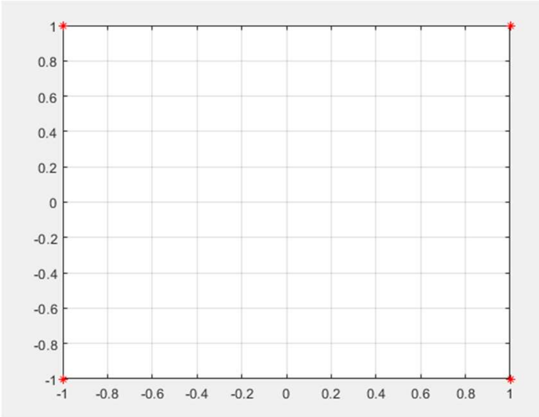

Figure (2): 4-QAM graph

Therefore, it has computed the IFFT and the cyclic prefix, this cyclic prefix has a length of 16 symbols and it assumed to be long enough (and the frame synchronization sufficiently accurate) so that the transmission is performed without inter-carrier and inter-symbol interference.

To finish with the emitter, it has made a parallel to a series to be able to transmit the data. To perform this operation, it has concatenated each column of the matrix into a vector.

Then it has passed the signal for the channel OFDM, it has used a MATLAB command for do this basic OFDM channel AWGN, this is a basic noise channel with average equal to zero and gaussian variance. But first it has done a loop for compute this for different values of SNR.

In the receiver it has done the same step that the sender but in the reverse order. First, it has done a serial to parallel then it has removed the cyclic prefix and hereafter the fft to switch to the frequency domain another again. Finally, we have demodulated the signal and it has recovered our signal.

It has computed the basic OFDM channel for different values of the SNR and it can observe in these graphs how the demodulator demodulates the signal if it has a lot of noise (bad SNR) or when it has low noise levels (good SNR).

It has generated these graphs for different values of SNR, for the left graph, it has used the worst SNR (in our case

-3 dB) and it can see the low precision and in the right graph, it has used the best SNR (in our case 10 dB) and it can see that despite the noise, the most of the received symbols is equal to the sent symbols.

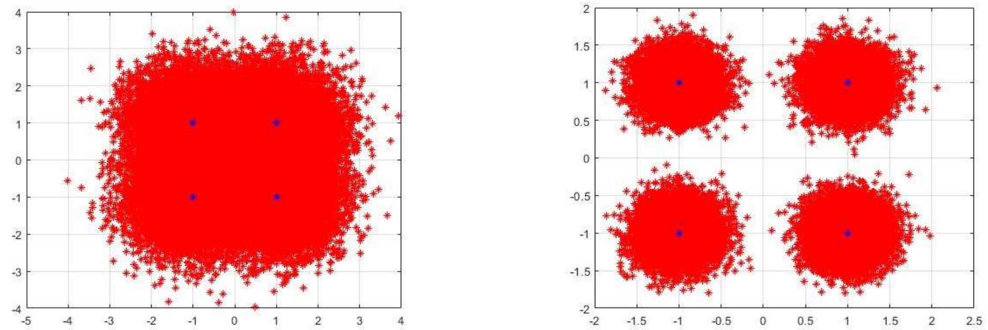

Figure (3): Decoding for different values of SNR

For finish this step it has computed the BER to be able to see graphically the curve of our bit error rate and to be able to compare it with the standard BER vs SNR.

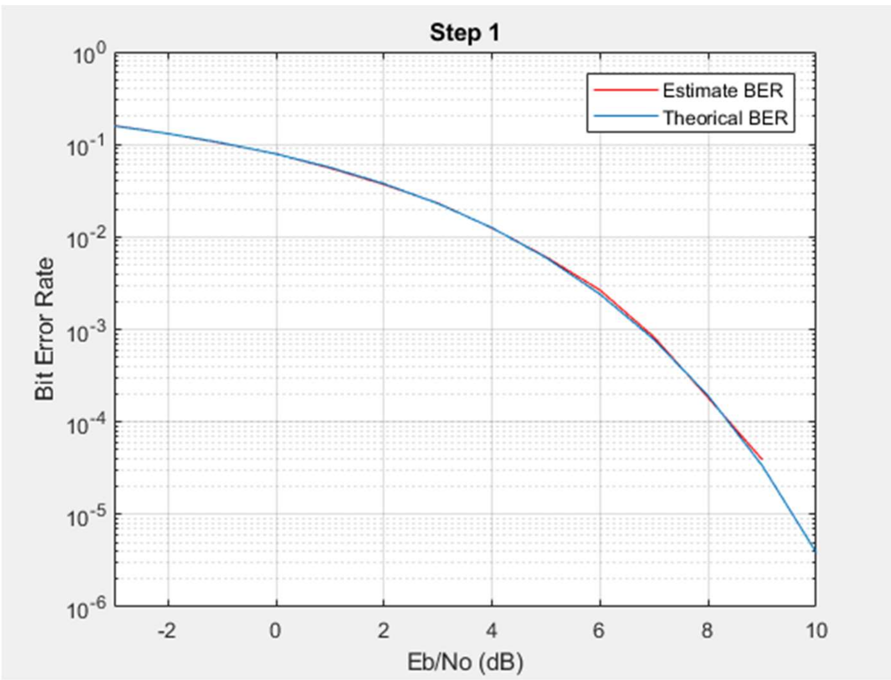

Figure (4): BER VS SNR graph

It has computed this step for a range to -3 and 10 dB because we would like to see how our model behaved for high noise levels. Our model is not equal to the theorical but it is very similar, this is possible because it has only used a basic OFDM channel. Our model has a little lines and there is not curved but the two curves are identical until the 4 dB.

Step 2 Resource Allocation

In general, a data rate can be adaptively varied with the channel variation. Such link adaptation is useful to maximize the system bandwidth efficiency. One particular example of link adaptation is time-domain AMC (Adaptive Modulation and Coding) technique, in which a multiple number of time slots with different channel gains are shared among the different users.

The resource allocation algorithm deal with how the resources like power and bit allocation can be done efficiently to improve the spectral efficiency and to achieve the maximum system capacity. Allocating the total available power and maintaining fairness among users is a difficult task. It is necessary to maintain tradeoff between system capacity, computational complexity and fairness. Several optimization techniques are used to minimize the overall transmit power of a multiuser OFDM system, but here, the underlying key principle in the frequency-domain AMC technique is the **water-filling algorithm**.

For this part, it is assumed a QAM modulation, and a target error probability in each subcarrier:

$$P_{e,tar} = 10^{-5}$$

It is also given the channel impulse response of the channel we are going to use:

$$h = [h(1), h(2), \dots, h(8)]$$

With the channel impulse response, the channel attenuation H in each subcarrier can be computed by doing a FFT of size equal to the number of subcarriers used ($N = 128$).

Consider an OFDM system with N subcarriers, each with a subcarrier spacing of Δf . The capacity of a subchannel corresponding to the k subcarrier, is given by the Hartley-Shannon channel capacity, such that:

$$C_k = \frac{1}{2} log_2 \left(1 + |H(k)|^2 \frac{\sigma_{x,k}^2}{\sigma_{n,k}^2}\right)$$

$$\sum_{k=1}^{N} \sigma_{x,i}^2 \leq P_{max}$$

where $|H(k)|^2$, $\sigma_{x,k}^2$ and $\sigma_{n,k}^2$ denote the frequency response, transmission power, and noise variance of the k subchannel, respectively. Then, the total channel capacity is given by the summation of the capacity for individual subcarriers, that is,

$$C = \sum C_k$$

The main problem is find the set of $\sigma_{x,i}^2 > 0$ that maximizes the capacity under the power constraint. Remark that $\sigma_{x,i}^2 = P_i$

To maximize the available bit rate, we first solve the power allocation problem. The objective is to maximize the capacity by allocating new powers to each subchannel:

$$C_{max} = max_{Pi} \sum_i \frac{1}{2}\, log_2 \left(1 + \frac{Pi}{\sigma_{n,i}^2}\right) \rightarrow \sum_i Pi \leq P_{max}$$

KKT conditions → $Pi = \begin{cases} Pi = \mu - \frac{No}{|H|^2},\ Pi > 0 \\ \frac{No}{|H|^2} > \mu,\ Pi = 0 \end{cases}$

Note that $\frac{No}{|H|^2}$ is the Noise to Carrier Ratio (NCR) for each subcarrier.

The power maximum P_{max} we are going to send is fixed at 1. It is important to take care on this constraint along the power allocation procesure, because the powers can change but the addition of this powers should not be higher than the total power.

We find the water level (μ) with an algorithm that suppose an initial μ:

$$\mu = \frac{Pmax + \sum_{i\epsilon I} \sigma_{n,i}^2}{|I|} \qquad I = \{i | \sigma_{x,i}^2 > 0\}$$

At first, all subchannels are supposed to have power higher to 0, and we find a first water level. Then, we calcule the power referred to each channel and we check if there are powers lower to 0:

1. If this power lower to 0 are find, we have to remove the worst channel and compute the new μ and the new Power vector.

2. If there are not powers lower to 0, the algorithm is finished and we have the right power allocation with the right tank water level. So, the capacity is maximum at this point:

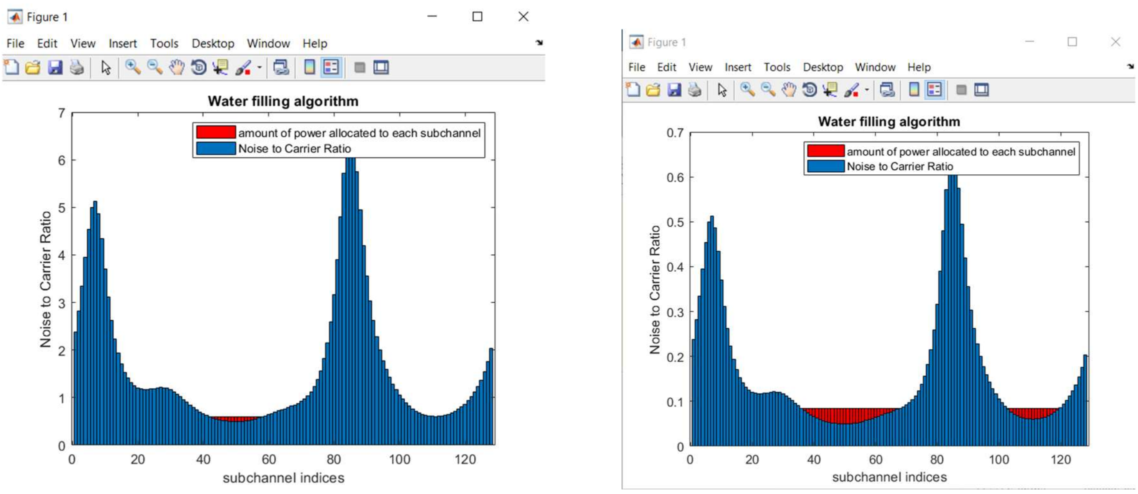

Figure (5): Water-filling algorithm for a SNR=0 dB and SNR=10 dB

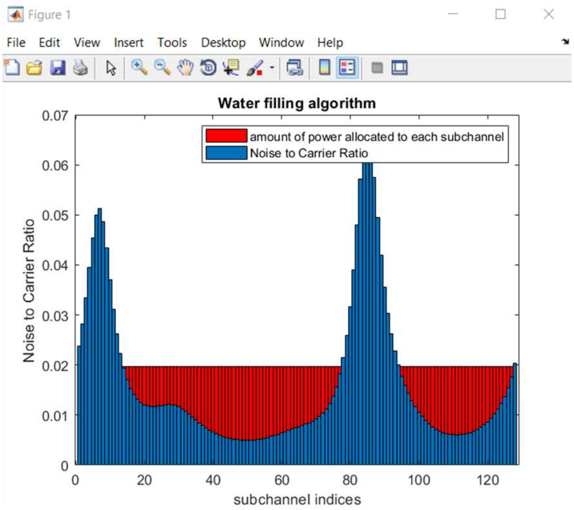

Figure (6): Water-filling algorithm for a SNR = 20dB

In these 3 graphics it can be observed the different level of the tank (different powers allocation) for different SNR. The conclusion we get when we use a higher SNR is that there are more channels used for do the power allocation, so we assigned new more powers to these subchannels and we get a higher capacity (there is a better optimization for more powers).

Finished the power allocation, we now have to deal with the bits allocation. The total data rate and BER in an OFDM system can be varied by the power allocation to the subcarriers. In other words, the data rate, transmission power, and BER are parameters that are associated with each other.

Hence, one or two parameters among these ones can be optimized, following the constraint associated with the rest of the parameters. If the transmission power and BER should be the same over all subcarriers, the number of bits that can be allocated to each subcarrier is given by:

$$b(k) = \frac{1}{2}\log_2\left(1 + \frac{SNR(k)}{\Gamma}\right)$$

Where $SNR(k)$ is the SNR refered to each subcarrier and Γ is the SNR gap. This gap is used to measure the reduction of SNR with respect to capacity and it only depends on target error probability.

As the QAM is the modulation taken for this part, we can compute the SNR gap recalling the error probability of a QAM transmission (on an AWGN channels) and inverting this expression to obtain the usable constellation size:

$$P_{e,QAM} \approx 2erfc\left(\sqrt{\frac{3 * SNR}{2(2^{2b}-1)}}\right) \rightarrow inverting \;\rightarrow b \leq \frac{1}{2}\log_2\left(1+\frac{3 * SNR}{2\left[erfc^{-1}\left(\frac{P_{e,targ}}{2}\right)\right]^2}\right)$$

If we compare with the capacity formula, we can easily find Γ as:

$$b \leq \frac{1}{2}\log_2\left(1+\frac{3 * SNR}{2\left[erfc^{-1}\left(\frac{P_{e,targ}}{2}\right)\right]^2}\right) \quad \longleftrightarrow \quad b = \frac{1}{2}\log_2\left(1+\frac{SNR}{\Gamma}\right)$$

$$\Gamma = \frac{2}{3}\left[erfc^{-1}\left(\frac{P_{e,targ}}{2}\right)\right]^2$$

Having the SNR gap we can compute the new bits allocation, and the resource allocation problem is solved:

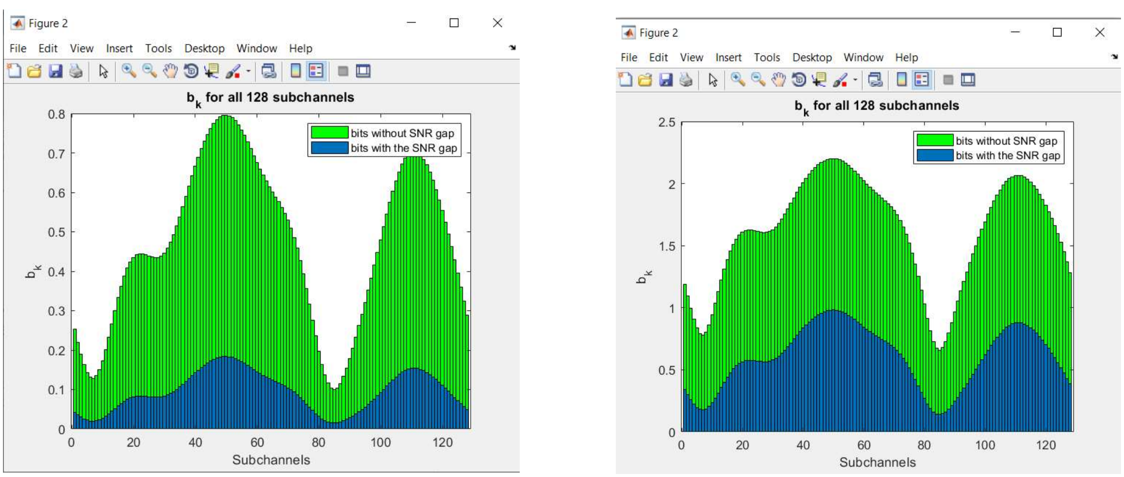

Figure (7): Bit allocation for a SNR = 0dB and SNR = 10 dB

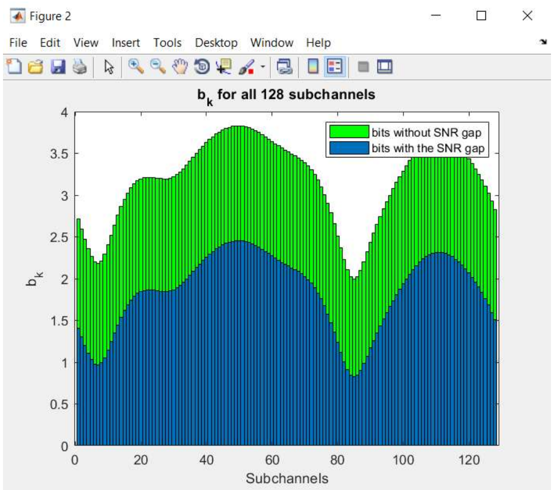

Figure (8): Bit allocation for a SNR = 20 dB

Step 3 Channel Estimation

The third step is focused on the cannel estimation in an OFDM transmission.

In an OFDM system, the transmitter modulates the message bit sequence into QAM symbols, performs IFFT on the symbols to convert them into time-domain signals, add the CP to avoid ISI and sends them out through a (wireless) channel. The received signal is usually distorted by the channel characteristics. In order to recover the transmitted bits, the channel effect must be estimated and compensated in the receiver.

Each subcarrier can be regarded as an independent channel and consider orthogonality among subcarriers. The orthogonality allows each subcarrier component of the received signal to be expressed as the product of the transmitted signal and channel frequency response at the subcarrier.

The transmitted signal can be recovered by estimating the channel response just at each subcarrier. In general, the channel can be estimated by using a preamble or pilot symbols known to both transmitter and receiver. In our case, this preamble is given by two OFDM symbols, I_k , sent on each subcarrier $k = (0,1,2,\dots,N-1)$:

$$I_k = (-1)^k$$

To introduce the pilots is used a block type of pilot arrangement. In this insertion type, OFDM symbols with pilots at all subcarriers are transmitted periodically for channel estimation. Since pilot tones are inserted into all subcarriers in a period of time, the block-type pilot arrangement is suitable for frequency-selective channels.

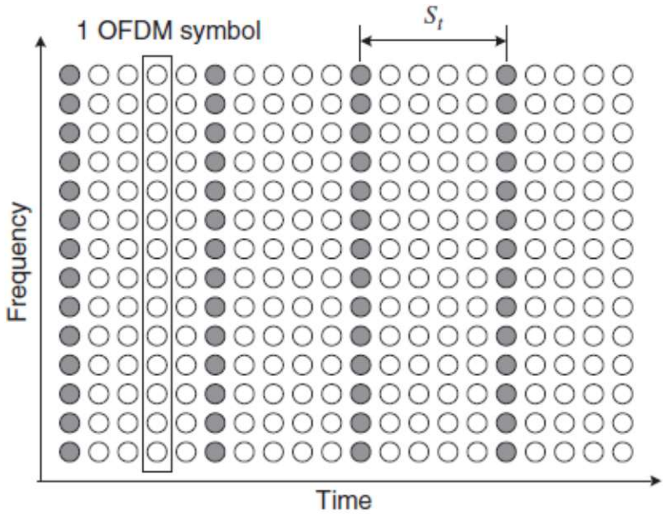

Figure (9): The insertion of the pilots

First of all, it is necessary to create a channel to compare the estimation with the real channel. In this part, it is assumed an 8-tap Rayleigh channel with uniform power delay profile. So, each tap has the same power that the next tap and the channel is computed as:

$$channel\ (h) = \frac{randn(1,8) + j * randn(1,8)}{\sqrt{2}}$$

After the channel definition, it is known the channel impulsive response, **h**, with 8 components (8-taps). Then a FFT is made to get the channel frequency response.

Having the channel, it is time to send the preamble. The pilot matrix is defined as:

$$X = \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \\ \dots \\ 1 \end{pmatrix}$$

We decided to introduce the pilots at the beginning of the OFDM matrix symbols. Like that, the firsts two symbols received can be used to estimate the channel.

A first pilot is sent over the 128 subcarriers, with the CP addition, and it is convolutioned by the impulse response before add the AWGN. At the receiver we remove the cyclic prefix and we get the signal received, **Y**.

Given the channel gain $\boldsymbol{H}$ for each subcarrier, the received matrix can be recovered:

$$Y = \begin{pmatrix} y(1) \\ y(2) \\ y(3) \\ \dots \\ y(128) \end{pmatrix} = \begin{pmatrix} x(1) \\ x(2) \\ x(3) \\ \dots \\ x(128) \end{pmatrix} \begin{pmatrix} H(1) \\ H(2) \\ H(3) \\ \dots \\ H(128) \end{pmatrix} + \begin{pmatrix} z(1) \\ z(2) \\ z(3) \\ \dots \\ z(128) \end{pmatrix} = XH + Z$$

To estimate the channel, there are two different methods: Least-Square (LS) error and Minimum Mean Square Error (MMSE)

In order to choose one, we decided to take the Least-Square error because is easier to adapt in MATLAB code.

The least-square (LS) channel estimation method finds the channel estimate H_{LS} in such a way that the following cost function is minimized:

$$J(H) = \left|\left|Y - XH_{LS}\right|\right|^2$$

$$H_{LS} = X^{-1}Y \;\rightarrow\; H_{LS}(k) = \frac{Y(k)}{X(k)}$$

The Mean Square Error (MSE) of this estimation can be calculated as:

$$MSE_{LS} = \frac{\sigma_z^2}{\sigma_x^2}$$

When the estimation is complete, the new channel impulse response, $h\_estimate$, can be calculated by doing an IFFT on the H_{LS} and taking the first 8 numbers. Finally, we apply the demapping without pilots and we demodulate the signal.

To calculate the MSE vs $\frac{Eb}{No}$ graphic, we have used an SNR array [0,20] dB:

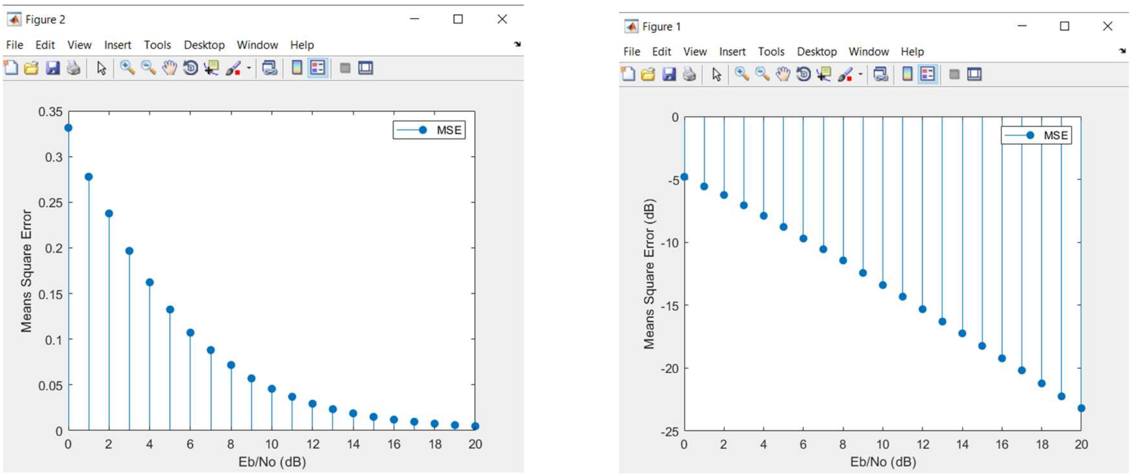

Figure (10): MSE graphs

On the left graphic there is a great reduction of the MSE when the SNR increases. So, if we want to improve our MSE and have a best estimation we only have to use a higher SNR.

By the way, the MSE can be improved too by sending more pilots. If we send more pilots, the channel impulse response estimation will be calculated as the mean of the different estimation:

$$h = \frac{\sum_{i=0}^{P} h(i)}{P}$$

where P is the number of pilots we sent, and so the differents estimations. The noise will be more reduced due to its value, that is similar in for all pilots:

$$No = \frac{\sigma_n^2 \sum_{i=0}^{P} 1}{P}$$

Step 4 Optimal Viterbi Decoding

In the step 4, It have supposed that the channel is not known at the transmitter, and it have used a convolutional coding to counteract the possibility of deep fades on some of the subcarriers.

- Code Rate =1/2
- Frame of Lf =128 bits
- The convolutional code: $G(D) = [1 + D \quad 1 + D + D^2]$.
- The 256 coded bits are multiplexed on the 128 subcarriers modulated in 4-QAM, and forming one OFDM symbol.

In addition, the receiver is either assuming perfect channel knowledge or using the channel estimation obtained in step 3. Finally, it considers Viterbi decoding using the soft decision in specific the Maximum Likelihood Sequence Detection (MLSD).

For the convolutional codes it has generated 128 random bits and it has used this expression to encode:

$$G(D) = [1 + D \quad 1 + D + D^2]$$

It knows that each bit encode depends on the las two bits that have entered to the encoder. We obtained 2 bits for each bits encode so, the first output bits depends only the last bits ($X_1 = 1 + D$) and the second output bits, depends only the las two bits ($X_2 = 1 + D + D^2$). With this expression it is generated a table for see who will be the next state and finally it has obtained two diagrams.

U_i	U_{i-1}	U_{i-2}	X_1	X_2	State	Next State	Decimal
0	0	0	0	0	a	a	0
1	0	0	1	1	a	b	3
0	0	1	0	1	b	a	1
1	0	1	1	0	b	b	2
0	1	0	1	1	c	c	3
1	1	0	0	0	c	d	0
0	1	1	1	0	d	c	2
1	1	1	0	1	d	d	1

Table (1): Trellis diagram

These states are the two last bits and it has represented with the letters: a =00 b=01 c=10 d=11

Thanks to this table these diagrams have been built, we can use any of them to obtain our output sequence, you just have to follow the path until encoding the 128 bits, then it will have 256 output bits.

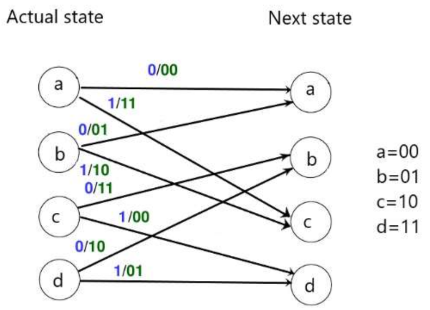

Figure (11): Trellis diagram

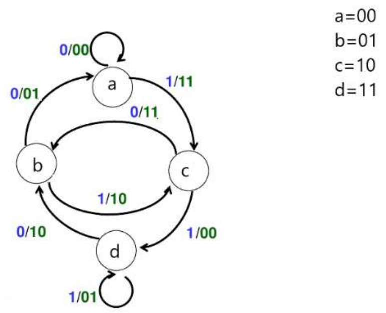

Figure (12): State diagram

In these diagrams it can see that if the last two symbols were 00, it is means the state a, and the next symbols is a 0, it remain at zero, the same state but it has a 00 in the output (the number in green). Using the diagrams for all the bits it can code all the frames.

When it has obtained the encoded sequence, it has done the same process that all the step. First it has done a 4-QAM modulation, it has transformed the two columns with 128 bits each column (X_1 and X_2) in one column with 128 numbers with real part and imaginary part. Then it has divided by $\sqrt{2}$ because it has transmitted power not energy.

In addition, it has done the fft, the addition of the cyclic prefix and the parallel to serial.

Therefore, it has passed the signal for the channel from the step 3, in this case it is charged the channel.
We have directly loaded the channel from step 3, to make sure we use it. We have also loaded the estimate of the channel for later use in the Viterbi decoding.

It has derived the expressions from the Maximum Likelihood Sequence Detection depending on the channel. It is the same that the slide but multiplying the inputs bits of the decoder by the frequency response of our channel. In this way it takes into account the channel knowledge and the specific gain in each subcarrier.

$$b(D) = argmax\ P[y(D)|b(D)]$$

$$b(D) = argmax\ P[y(D)|b(D)] = argmin|y(D) - b(D)|^2$$

$$argmin|y(D) - b(D)|^2 = \sum_{j=1}^{n}\sum_{i=0}^{L-1}|y_{ji} - b_{ji}h_i|^2$$

Finally, we should include here our Viterbi graph and explain it but unfortunately, we have not been able to implement the Viterbi decoding in MATLAB, we have tried to perform the algorithm but we have a failure and we could not to get the best path. We have had a lot of programming problems because we have not had the level of programming.

Conclusion

The aim of this project was to understand an OFDM transmission and the behavior of the channel the signal will pass for. First, we have design the basic OFDM channel and then we have supposed the different situation about the channel: when is known both transmitter and receiver, and when we have to estimate it.

Thanks to the project, we have understood how an OFDM transmission can be improved by allocating new powers to each subchannels (and new bits too), and we have understood too the Viterbi algorithm to deduce the frame that was sent at the beginning with the least error possible.

In addition, this project has been useful for us in particular to get into the programming world, knowing the different function we can define and using the programme language (MATLAB) for create our proper transmission, which we have to know if we study Telecommunication.

What is the application of the OFDM modulation? We have searched some information about it and the first application is the 4G, so we now know how to implement a modulation that works for the LTE technology.

Even if we did not finish the Viterbi decoded, we really know how it works and what is used for. It has been hard to implement the what you think in the programme, but we are satisfied with the work we have done and the new knowledges about an OFDM transmission.