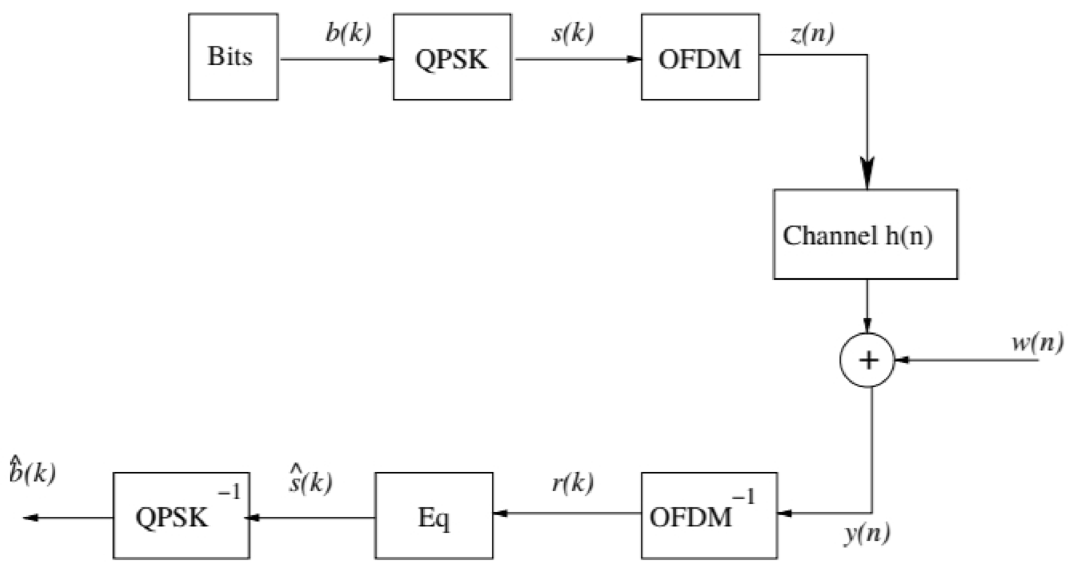
SSY130 - Project 1A
Baseband Communication

Diego Jauregui   Lucas Rath   Sondre Wiersdalen   Mohammad Shaikh

November 20, 2018

student id: 19930425
secret key: Jumpluff



1 Part A: Baseband Communication

Question 1) Default case: ideal channel, no noise added, $N_{cp} = 0$, know channel

(a) The ideal channel consists of a Kronecker delta function $h = \delta_0(n)$, which in frequency domain results in $H(\omega) = 1$. Therefore, the channel does not modify the signal $z(n)$ and for the cases where we do not have any noise, $z(n) = y(n)$.

The equalization will then convolute the distorted signal $r(k)$, removing the effect of the channel. However, since $H(\omega) = 1$, there is no effect to remove and therefore $\hat{s}(k) = r(k)/H(k) = r(k)$. As a result, the transmitted symbols are exactly equal the received symbols.

Furthermore, the errors related to the message transmitted are EVM: 2e-16, BER: 0, confirming that there were no errors bigger than machine precision and the transmitted symbols are correct.

(b) The cyclic prefix is essential for the proper operation of out communication scheme to work reliably because it acts as a guard region preventing interference between two sequence of OFDM symbols that might occur during the channel convolution.

In order to avoid completely this inter-symbol interference the cyclic prefix must be at least of the same size of the discrete impulse response of the channel $N_{cp} \geq N_h - 1$, where N_h is equal to the number of samples needed to the channel input response reach a "stationary" behaviour with value very close to zero.

In this way, the interference caused by the channel convolution of the last block will be entirely absorbed by the prefix samples, that will be discarded later by the receiver [Oltean and Naforniţă, 2003].

Another benefit of using cyclic prefix is that using this technique, we get a convoluted signal $y(n)$ that is exactly the same to the case when the transmitter was turned on at n=-∞. In this way, we can use the equalization procedure for periodic signals, which is faster since it makes operations using a DFT of length N samples instead of N+N_h-1.

(c) Both channels h_2 and h_3 consist of a Kronecker delta function $h = \delta_0(n)$ scaled by a factor. In other words, the channel convolution does not change the "dynamics" of the signal but only shifts the phase and/or scale the module. In frequency domain, those channel impulse responses can be described as:

$$\begin{aligned} h_2 &= 0.5 \cdot \delta_0(n) &&\Rightarrow H_2(w) = 0.5 \\ h_3 &= exp(0.5i) \cdot \delta_0(n) &&\Rightarrow H_3(w) = exp(0.5i) = 1 \underline{/0.5} \end{aligned} \tag{1}$$

Therefore, α is equal 0.5 for h_2 and exp(0.5i) for h_3. We show then in figure 1 the pre-equalization symbol constellation diagram for both channels:

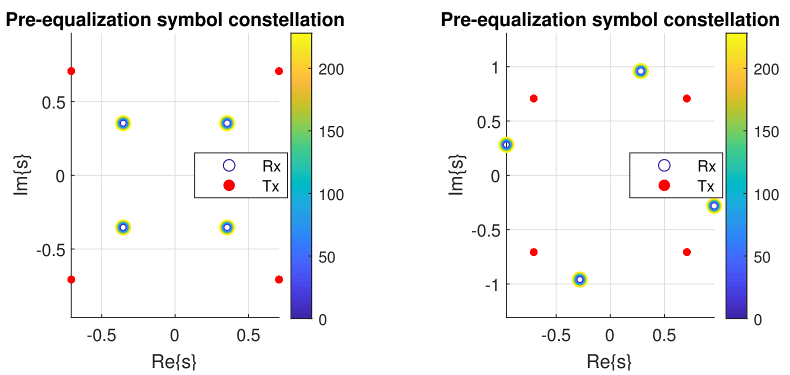


Figure 1: *Pre-constellation for channel $h_2(n)$ on the left and $h_3(n)$ on the right side.*

The post-constellation graphs are not displayed here but the transmitted and received signal matched perfectly for both cases, so that BER=0 and EVM=3e-16. As expected, the h_2 channel scaled the module of the symbols by 0.5, while the h_3 channel shifted the phase by 0.5 rad ≈ 28 deg.

(d) We now simulate the effect of a synchronization error when receiving the message. Introducing an delay of 1 time step we get errors EVM:1 and BER:0.5.

Curiously, the received message is correct in the beginning and in the end but completely messed in the middle. To investigate this effect, first we recall that this introduced synchronization error is equal to an unexpected delay of the signal. In a frequency perspective:

$$x(n-k) \quad \rightarrow \quad X(w) \cdot exp(-j \cdot w/w_s \cdot 2\pi \cdot k) \tag{2}$$

where k is the delay in time step units. As can be noticed, this delay introduced a distortion in the frequency domain, shifting the phase of $X(w)$ in $-(w/w_s \cdot 2\pi \cdot k)$ rad. Notably, this will have very little effect as $w \rightarrow 0$ or $w \rightarrow w_s$ since the phase shift will tend to zero.

We also know that the received data in frequency domain compose the message, so that the phase of each frequency determines our symbols, which are then ordered by growing frequency from 0 to w_s. Consequently, the initial and last symbols will be very little effected by the distortion. For this reason, the message is mostly corrupted in the middle, while the beginning and the end are still correct. The more we increase the delay k, the more distortion we will introduce in (2) and more effect we will cause in the initial and last symbols.

(e) Now we investigate the effect of adding noise when transmitting the message by changing the effective signal to noise ratio (SNR).

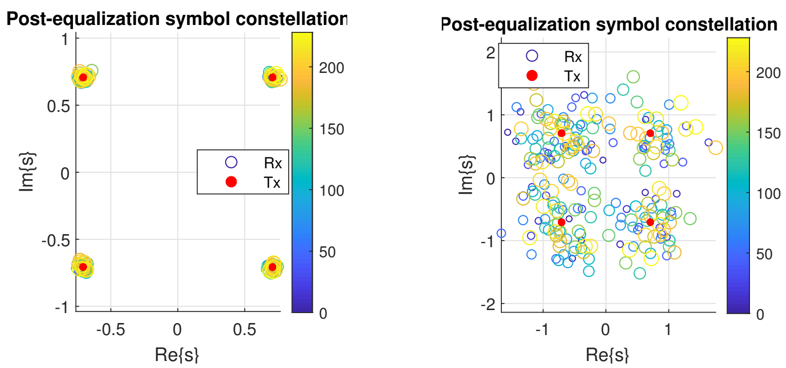


Figure 2: *Post-constellation for channel $h_1(n)$ with snr=30dB on the left and snr=5dB on the right side. Tx and Rx are the transmitted and received symbols respectively.*

As can be seen in figure 2 the more noise we add the more corrupted are the symbols. Indeed, one symbol is labeled correctly when after equalization it appears in its correct quadrant in the complex Cartesian plot. In other words, if the noise affects the received symbols but they still remain in their right quadrant, so the final message will not be distorted. For snr=30dB the introduced noise was not enough to corrupt the message, so it was transmitted without miss-classifications. On the other hand, for snr=5dB we added so much noise that some symbols moved to the wrong quadrant, resulting in the following communication (EVM:0.5, BER:0.02):

Transmitted:	`'Alice: Would you tell me, please, which way I ought to go'`
Received:	`'Alice: Wo}ld you ]d,l me, please, whic_ way I\$ouoht to go'`

Question 2) channel model to h_4 (the low-pass system) and set the cyclic prefix to 60.

(a) Since we do not have noise in the system, we know that $r(k) = H(k) \cdot s(k)$. For this reason, the vector $r(k)$ will be a simply element-wise multiplication of $H(k)$ and $s(k)$. It follows then that $r(1) = s(1) \cdot H(1) = \sqrt{1/2}(-1+i) \cdot 5 = -3.5355 + 3.5355i$, so that $r(1)$ can be found in the constellation diagram.

We can confirm that relation by checking that symbols at the beginning and at the end of the message present almost no phase deviation since $\angle H(w \rightarrow 0) \approx 0$ and module amplification close to 5, since $|H(w \rightarrow 0)| \approx 5$.

(b) When the system is in this setup, the value of EVM=1e-15 and BER=0. We notice that for all $N_{cp} \geq 59$ EVM remains approximately zero. On the other hand, if we choose $N_{cp} < 59$ we get some considerable error EVM that starts to increase as we keep reducing N_{cp}. However, even setting $N_{cp} = 0$ is not enough to get miss-classification of symbols, so BER=0.

We conclude then that the magic number is $N_{cp} \geq 59$, which is exactly the size of the channel impulse response minus one, such that $N_{cp} \geq N_h - 1$. This size of cyclic prefix is the minimum size so that the finite transmitted signal $z(n)$ looks like being periodic.

It is also important to notice that if we are transmitting two consecutive OFDM blocks, the cyclic prefix area will also absorb the convolution interference of the channel h between the blocks.

(c) The value of N_{cp} is of high importance when using the channel $h_4^{'}$. If we use $N_{cp} \geq N_h - 1 = 59$ we get zero EVM and BER. As we keep reducing the value of N_{cp} bellow 59, we starts to get considerable EVM error, until the point we start to get bit errors BER.

Since we are doing equalization on only N samples instead of N+M-1, we need the cyclic prefix to make the finite transmitted signal looks like being periodic. That is why we get errors when we do not use cyclic prefix using this channel.

Question 3) Realistic case: ideal channel, no noise added, $N_{cp} = 0$, unknown channel

(a) In the unknown-channel scenario the effect of the nonzero sync error is less destructive for the message than in the know-channel scenario. This phenomena happens because the receiver uses the pilot frame to estimate the channel impulse response during the transmission of the the data frame. In addition, the pilot also captures this effect of synchronization delay in time domain. Therefore, during the equalization of the unknown channel, we remove the effect of the channel convolution as well as the effect of synchronization delay.

(b) We need $N_{cp} \geq 59$ to obtain EVM=10^{-14}. This value is the same value we obtain in the know-channel scenario (question 2c). We obtain the same value for the N_{cp} in both scenarios because this is the minimum size that will avoid inter-symbol interference between the pilot and the data and also the minimum size that makes both frames look like being periodic signals.

Running some experiments, we noticed that this setup is more sensitive to noise level than the know-channel scenario. This happens because the system identification of the channel will also capture the noise introduced in the system:

$$\hat{H}(k) = \frac{r_p(k)}{s_p(k)} = \frac{s_p(k) * H(k) + w(k)}{s_p(k)} = H(k) + w(k)/s_p(k) \tag{3}$$

The equalization will then transmit this noise again to the final symbol, increasing then the error.

$$\hat{s}(k) = \frac{r(k)}{\hat{H}(k)} = \frac{s(k) * H(k) + w(k)}{H(k) + w(k)/s_p(k)} \tag{4}$$

(c) Observing the results we confirm that the cause of getting some bit errors for the multi-path channel h_5 is because its impulse response has some periodic frequencies where $||H_5(w)|| = 0$. For this reason, the convolution $r(k) = H(k) * z(k)$ will then lose information in these mentioned frequencies. In addition to that, the equalization procedure will not be able to recover the data with zero magnitude and therefore $\hat{s}(k) = r(k)/H(k) = 0/0$, leading to bit errors for characters that correspond to those sub-channel frequencies.

The critical difference between h_5 and h_5' is that the last one does not present any magnitude of zero in the frequency domain. In this way, the equalizer can remove the effect of h_5' and recover all the information, avoiding completely the bit errors.

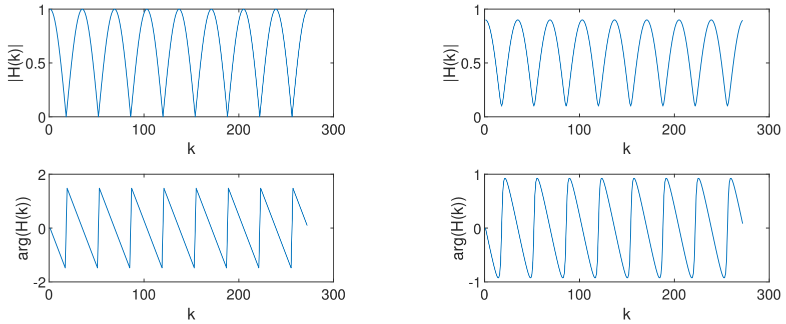


Figure 3: *Frequency response of* $H_5(k)$ *on the left and* $H_5'(k)$ *on the right*

(d) After playing around with the simulation, we focus in the BER as indicator of robustness against disturbance. We make the analysis to the three possible types of interference that we know so far (h_5 is plot but it is exclude of this analysis due to the inner error of its response):

ISI - We should not have problem with this if N_{cp} is chose wisely according channel response. Even if we don't choose an enough large value for N_{cp} the BER is not to large so in this context the received message will be very similar to the transmitted one. We can conclude that the unknown-channel scheme is robust against ISI. We obtain a BER of 6% using a channel with a response equivalence to Gaussian noise (h_6).

Noise (SNR) - For all the channels, the effect of the noise start to notice with a $SNR \geq 15$ for the ideal channels even with the phase shift (h_1, h_2, h_3) with a peak of approximate 8% in $SNR = 5$. The LP model (h_4) have the lowest tolerance to noise with a 25% at $SNR = 5$. Even these results, in general terms this scheme has a better tolerance that the known-channel scheme.

Synchronization error (early/late frame sync) - In this scheme we observe a favorable behavior with respect to the unknown-channel scheme. We don't observe the dead range now, instead we obtain a few errors in some characters which translate in a very low BER (less than 5% with $sync - err = 20$). This could be explain because the delay is affecting mostly the pilot frame, messing around the identification of H and consequently affecting the equalization of the data frame.

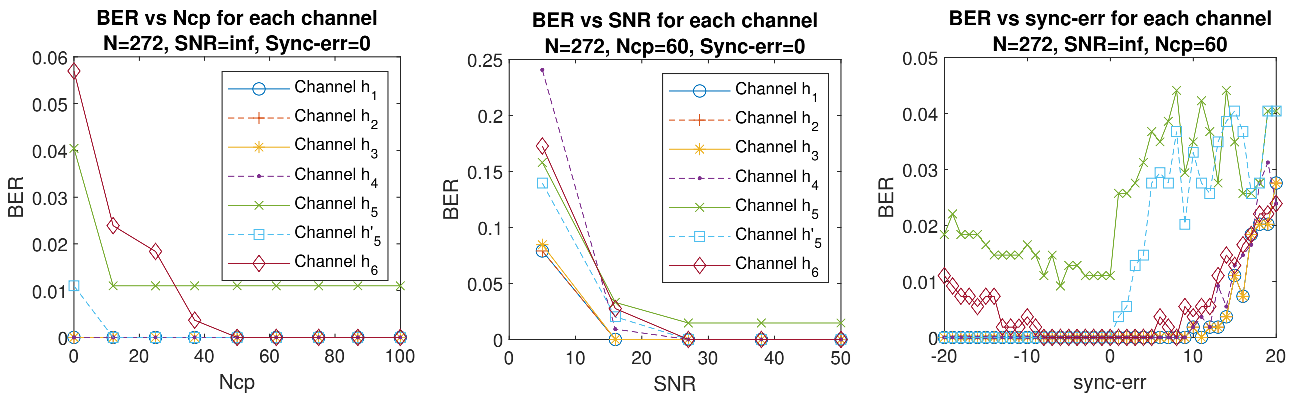


Figure 4: *BER vs* N_{cp}*,* SNR*,* $Sync - err$ *plots*

References

[Oltean and Naforniţă, 2003] Oltean, M. and Naforniţă, M. (2003). The cyclic prefix length influence on ofdm-transmission ber. *tom*, 48:62.