Elec-H-401: Modulation and Coding

Gauthier Duchêne – Nathan Dwek – Jason Rosa

Contents

1 Optimal communication over the ideal channel **2**
1.1 Halfroot Nyquist filtering 2
1.2 Impact of the noisy channel 2
1.3 Questions 4
1.3.1 Simulation 4
1.3.2 Communication System 5

2 Time and frequency synchronisation **7**
2.1 Effect of the CFO-caused ISI on the BER 7
2.2 Effect of uncompensated CFO on the BER 8
2.3 Sample time shift error 8
2.4 Sample time shift error correction using the Gardner Algorithm 8
2.5 Time of Arrival and CFO Estimation Using Frame Acquisition 11
2.6 Questions 12
2.6.1 Simulation 12
2.6.2 Communication System 18

3 Low-density parity check code **19**
3.1 LDPC encoding and hard decoding 19
3.2 Soft decoding 20
3.3 Questions 20
3.3.1 Simulation 20
3.3.2 Communication system 21

List of Figures **23**

1 Optimal communication over the ideal channel

The goal of this first section is to implement a minimal working simulation of a QAM communication. In order to achieve that, the up- and downsampling blocks, the RRC filtering blocks and the baseband equivalent model of the channel, as represented in figure 1 must be implemented. The modulator

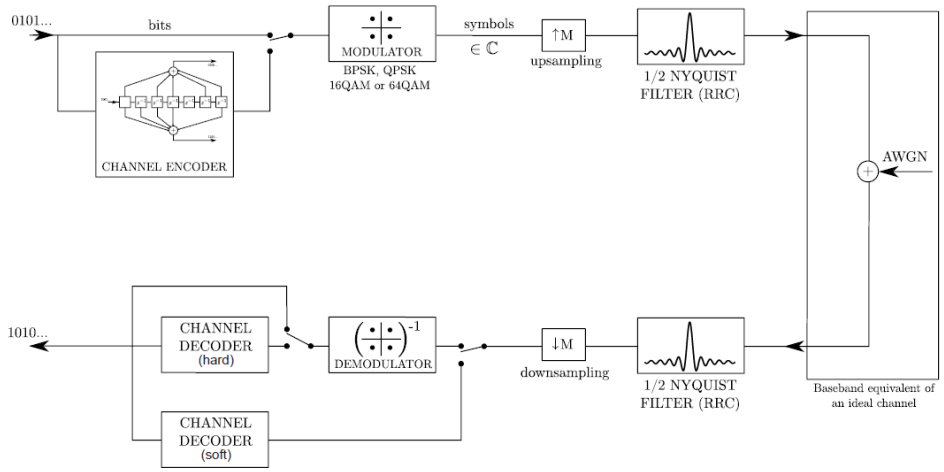

Figure 1: Block diagram of the communication system. [Source: Assignment introduction]

and demodulator are supplied with the assignment statement.

1.1 Halfroot Nyquist filtering

After its modulation, the message is upsampled and filtered with a root raised cosine filter to limit its bandwidth occupation. The effect on the PSD of the signal is shown in figure 2. The figures presented hereafter were produced using those same values for β, n_{taps} and f_m.

In order to maximize the SNR at the output, the low pass filtering is split between the transmitter and the receiver. The halfroot nyquist filter $g(t)$ is such that the resulting operation $h(t) = g(t) * g(t)$ forms a nyquist filter which does not introduce inter symbol interference, as shown in figure 3.

1.2 Impact of the noisy channel

Theory shows that a channel affected by AWGN can be modelled in the baseband by AWGN of corresponding power. This allows to easily simulate

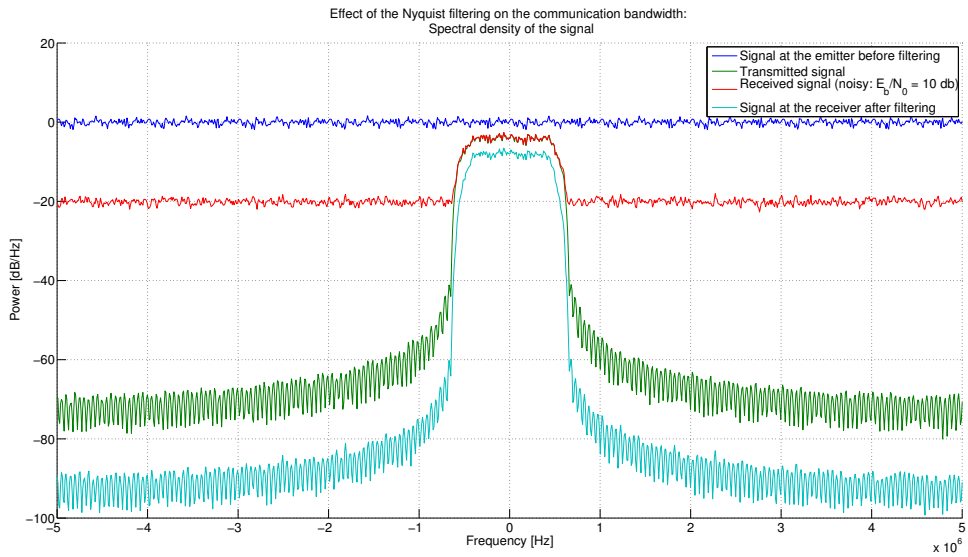

Figure 2: Nyquist filtering limits the communication bandwidth. $\beta = 0.3$, $n_{taps} = 20$ $f_m = 1\,\text{MHz}$.

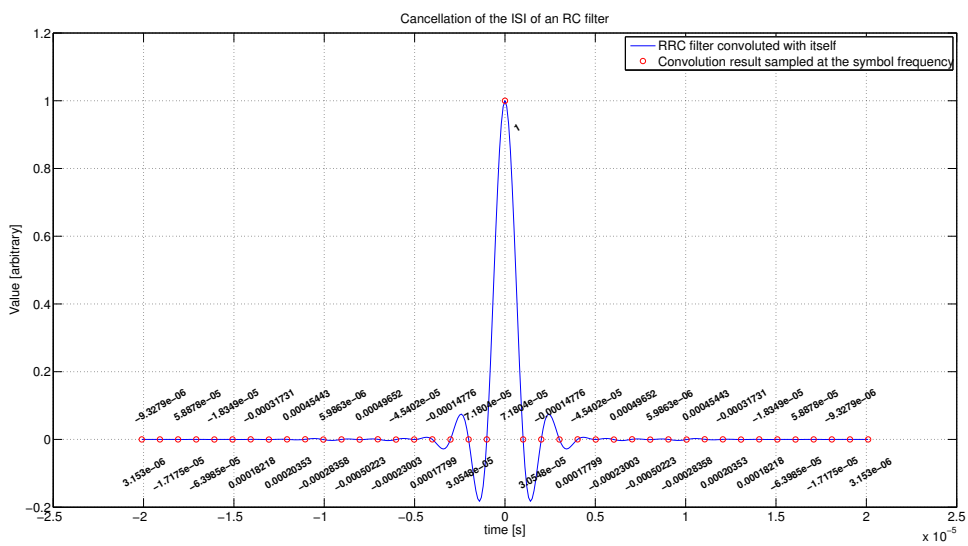

Figure 3: Cancellation of the inter symbol interference of a raised cosine filter.

the BER of a noisy channel. The results of the simulations are summarized by the BER curves of figure 4. Unless mentionned otherwise, the figures

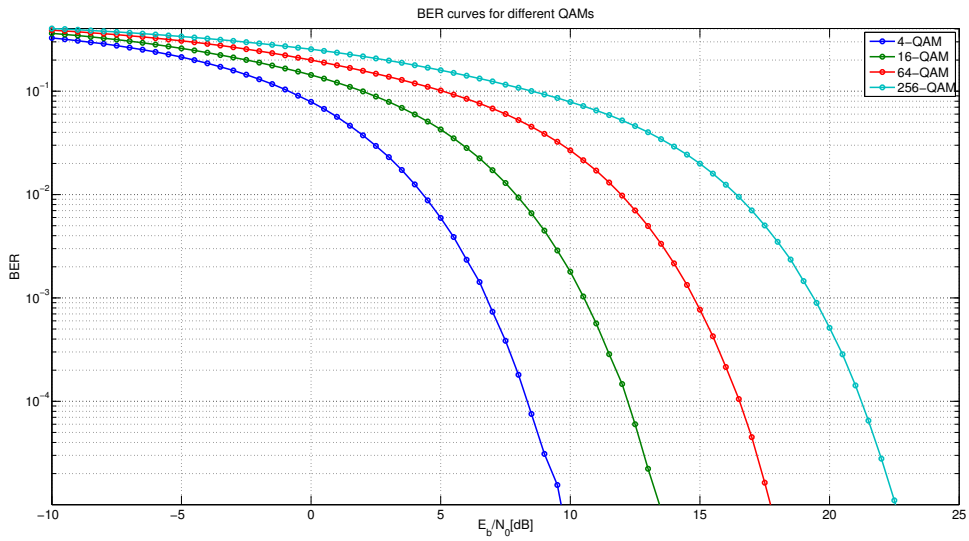

Figure 4: BER in function of $\frac{E_b}{N_0}$ for different QAM modulations.

presented in the remain of this report were produced using 16-QAM.

1.3 Questions

1.3.1 Simulation

It is proposed to use the baseband equivalent model of the AWGN channel. Would it be possible to live with a bandpass implementation of the system? Simulating in the baseband has the advantage of reducing the sampling frequency needed (at least roughly twice the carrier frequency if simulating in the passband, which is unrealistic for modern GHz links) and allowing to implement and simulate modulation and demodulation techniques regardless of this carrier frequency. This is why the baseband equivalent model is always preferred to simulate wireless communication channels.

How do you choose the sample rate in Matlab? To be able to observe the effects of Nyquist filtering, the sampling rate must be at least twice as high as the symbol frequency. To simulate sample time shift, even higher sampling rates should be used.

How do you make sure you simulate the desired $\frac{E_b}{N_0}$ ratio? Rather than adding a predetermined amount of noise to the signal, we first estimate its power in the useful frequency band, and then choose the noise power in order to obtain the required $\frac{E_b}{N_0}$ ratio. The baseband AWGN is given by:

$$n[k] = \sigma \cdot (\mathcal{N}(0,1) + j\mathcal{N}(0,1))$$

Where σ is given by:

$$\sigma = \sqrt{N_0 \cdot f_s}$$
$$N_0 = \frac{1}{2n_{bits}} \cdot \sum_k T_s \big|s[k]\big|^2 \cdot \frac{1}{\frac{E_b}{N_0}^* [\text{lin.}]}$$

Where $s[k]$ is the simulated signal at the emitter and $\frac{E_b}{N_0}^*$ is the desired $\frac{E_b}{N_0}$ ratio.

How do you choose the number of transmitted data packets and their length? In order to reliably observe a BER of 10^{-n}, common practice is to send 10^{n+1} bits of data. We can send those in one simulation because the whole system is time invariant and the noise is ergodic, which means that it is possible to draw the same conclusions from the observation of one long realization as from the observation of several shorter realizations. Additionally, the total number of bits should be a multiple of the number of bits per symbol for the modulator and the demodulator to work properly.

1.3.2 Communication System

Determine the supported (uncoded) bit rate as a function of the physical bandwidth. From the theory, we know that $R = \frac{\log_2(N_{sym})}{T}$ with R = bit rate, N_{sym} = number of symbols and $T = \frac{1}{f_m}$ = symbol duration. With Nyquist filtering: we know that $\Delta_f = 2f_m$, as shown in 2. Combining the two gives $R = \frac{\log_2(M) \cdot \Delta_f}{2}$

Explain the trade-off communication capacity/reliability achieved by varying the constellation size. As the constellation size increases for a given SNR, the distance between the "edges" of the constellations spots that are spread out by the noise decreases, which means overlaps and thus errors get more and more frequent. However, as seen in the previous question, increasing the number of symbols increases the bit rate for a given bandwidth.

This means that for a given SNR, it is possible to trade off reliability (higher BER) for capacity by increasing the number of symbols. Equivalently, it is possible to adapt the constellation size to the measured SNR to optimize the channel capacity given a specified BER, as is done in DVB-S2.

Why do we choose the halfroot Nyquist filter to shape the complex symbols? Filtering is required at the emitter in order to limit the bandwidth used by the transmission. However, ideally, filtering is also needed at the receiver in order to maximize the SNR, since the external noise can affect all frequencies. This is why we choose to split the filtering operation between the emitter and the receiver. Finally, ISI cancellation is required in order to be able to demodulate the signal properly. This is achieved by splitting a raised cosine filter between the emitter and the receiver.

How do we implement the optimal demodulator? Give the optimization criterion. The optimal demodulator is implemented using a matched filterbank $h_i(t)$ sampled at the symbol frequency. If the incoming signal $s(t)$ is matched to the filter and the sampling is correctly synchronized, $h_i(t) = s(-t)$ and the sampled output is a one since it is then equal to the autocorrelation of the $s(t)$ evaluated at 0 (no lag).

This output is disturbed by the correlation of the incoming noise with $h_i(t)$. Matched filters have the property that they maximize the SNR at $\frac{\mathcal{E}}{2N_0}$, which is the optimization criterion for a demodulator.

How do we implement the optimal detector? Give the optimization criterion. There are two possible optimization criteria:

- The maximum a posteriori criterion:
  "What is the most probable emitted symbol given the received signal?"

$$\underline{\tilde{s}}_m^{MAP} = \max_{\underline{s}_m} p[\underline{s}_m|\underline{r}]$$

- The maximum likelihood criterion:
  "What symbol is the most likely to have produced the received signal?"

$$\underline{\tilde{s}}_m^{ML} = \max_{\underline{s}_m} p[\underline{r}|\underline{s}_m]$$

The MAP criterion minimizes the BER, but requires to know the a priori probabilities ($p[\underline{s}_m]$), which is not practical. Fortunately, if the message is

correctly entropy coded, the M transmitted symbols are equiprobable and the two criteria are equivalent. The QAM detector implements the ML criterion by selecting the "allowed" symbol which minimizes the euclidean distance to the received symbol.

2 Time and frequency synchronisation

In a practical application, the emitter and receiver do not share the same physical location and use local oscillators. For this reason, the communication is affected by different effects: CFO and SCO due to the imperfections of the crystals, carrier phase error and sample time shift due to the fact that the oscillators are not physically the same. The baseband model of the effects of these imperfections is presented on figure 5.

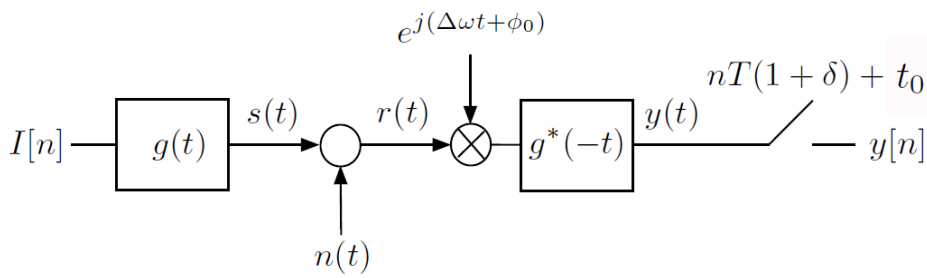

Figure 5: Baseband model with synchronization errors. [Source: course notes]

In this section, we will implement estimation and compensation algorithms for the sample time shift and the CFO, then assess their performances.

2.1 Effect of the CFO-caused ISI on the BER

The degradation of the BER due only to the ISI introduced by the CFO is shown in figure 6. Figure 6 shows that for a given value of $\Delta\omega$ and at higher noise levels, the BER curve is worsened but shows similar behaviour as the original curve with no CFO. However, past a certain value of $\frac{E_b}{N_0}$, the BER stops decreasing and only the errors introduced by the ISI subsist. Those plateaus in the curves correspond to what we observed when we were testing the simulation with no CFO but with a badly defined pair of nyquist filters which did not sufficiently cancel the ISI.

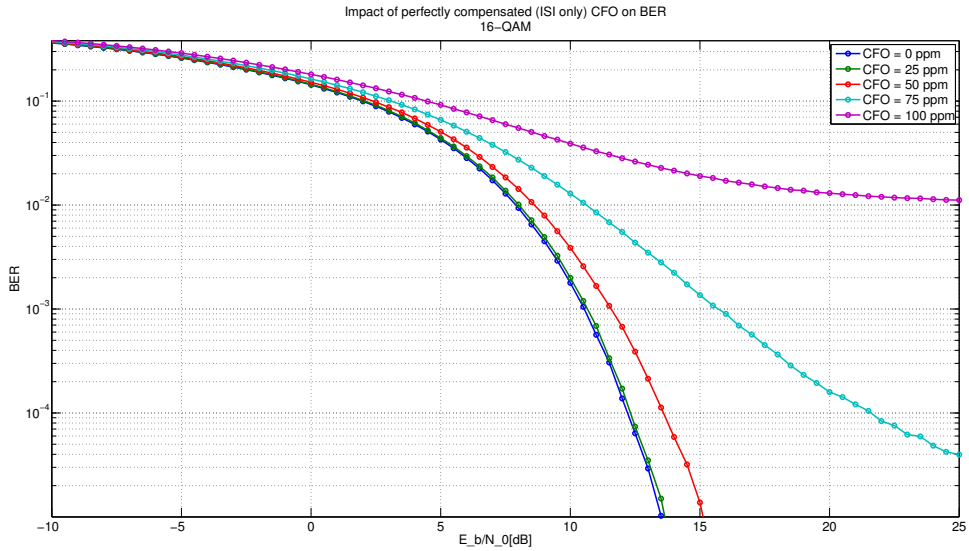

Figure 6: Impact of the ISI introduced by the CFO on the BER.

To conclude, figure 7 shows the ISI of two halfroot nyquist filters separated by CFO.

2.2 Effect of uncompensated CFO on the BER

If the CFO is not compensated, the slow phase drift over time renders the channel unusable and the BER is 0.5 at any noise level. Figure 8 shows the effect of noise and CFO on the constellation. This helps understand how uncompensated CFO renders any communication impossible.

2.3 Sample time shift error

Figure 9 shows the impact of the sample time shift on the error rate. As expected, the error rate is worsened by the sample time shift and the channel is unusable when the incoming signal is sampled exactly in between symbols.

2.4 Sample time shift error correction using the Gardner Algorithm

Figure 10 shows the convergence of the Gardner algorithm for different error weights. As predicted by the theory, the algorithm converges faster for higher values of κ, but at the expense of accuracy and stability.

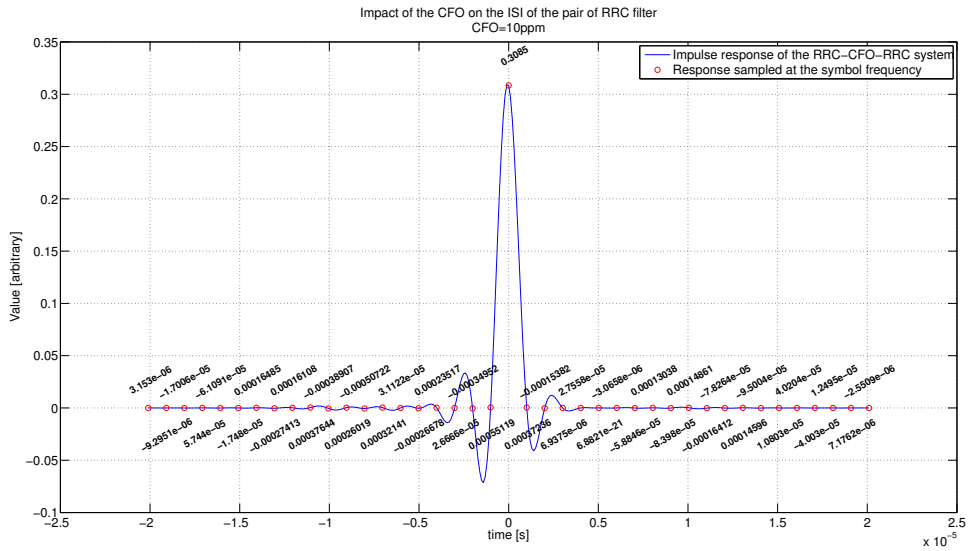

Figure 7: Effect of 10 ppm of CFO on the ISI of the pair of filters. $f_c = 2\,\text{GHz}$, $\Rightarrow \Delta\omega = 20\,\text{kHz}$.

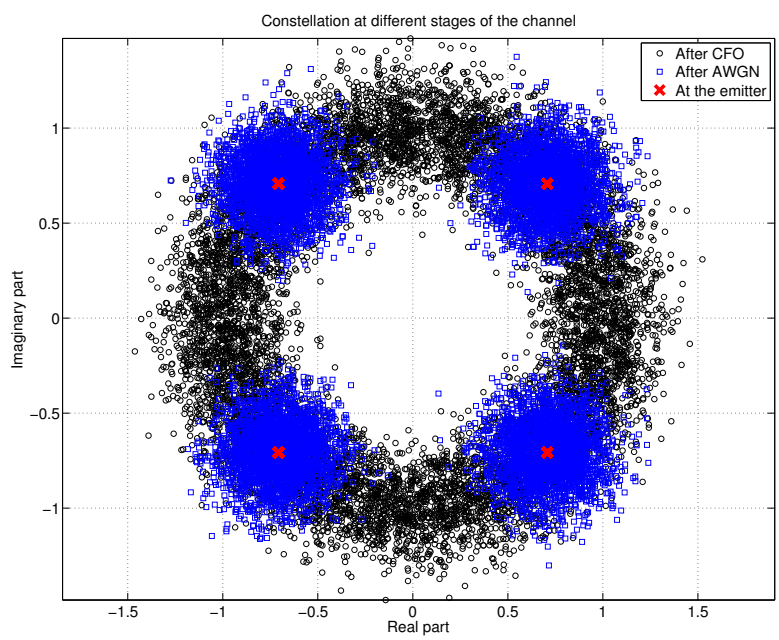

Figure 8: Constellation at different stages in the communication. 10 ppm of CFO, $f_c = 2\,\text{GHz}$, $\frac{E_b}{N_0} = 10\,\text{dB}$.

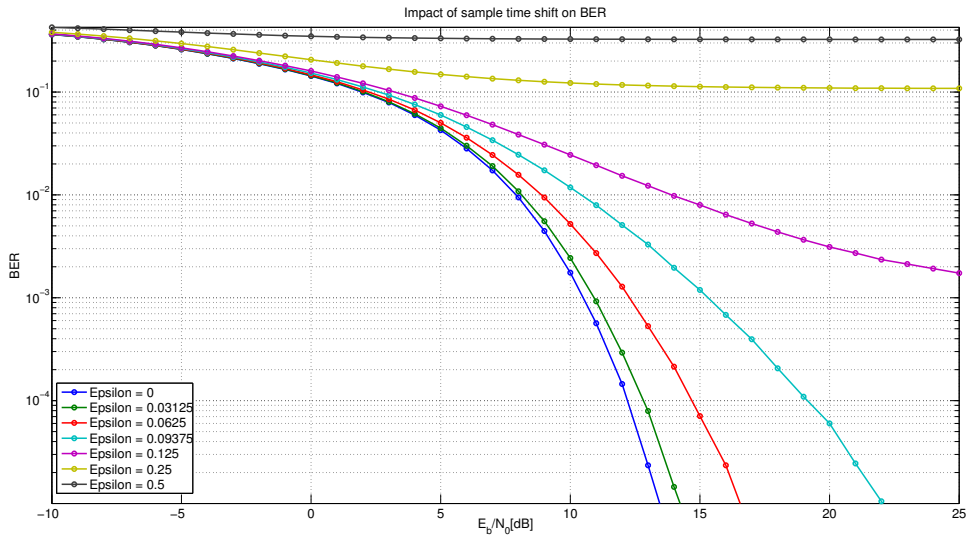

Figure 9: Impact of the sample time shift on the error rate. $f_m = 1\,\text{MHz}$, $f_s = 32\,\text{MHz} \Rightarrow \Delta k = 0, 1, 2, 3, 4, 8, 16$ samples.

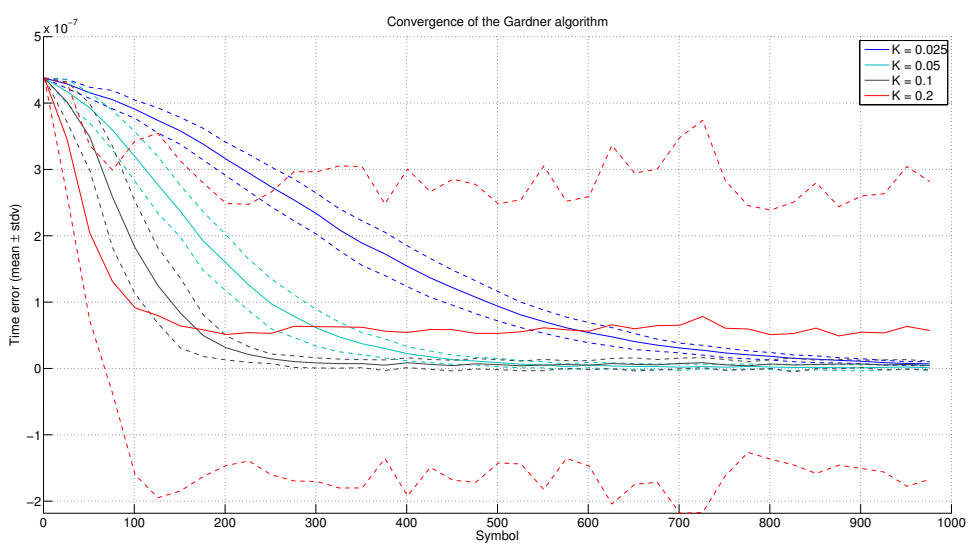

Figure 10: Convergence of the algorithm as a function of κ. $\frac{E_b}{N_0} = 10\,\text{dB}$.

The resistance of the algorithm to CFO is shown in figure 11. We see

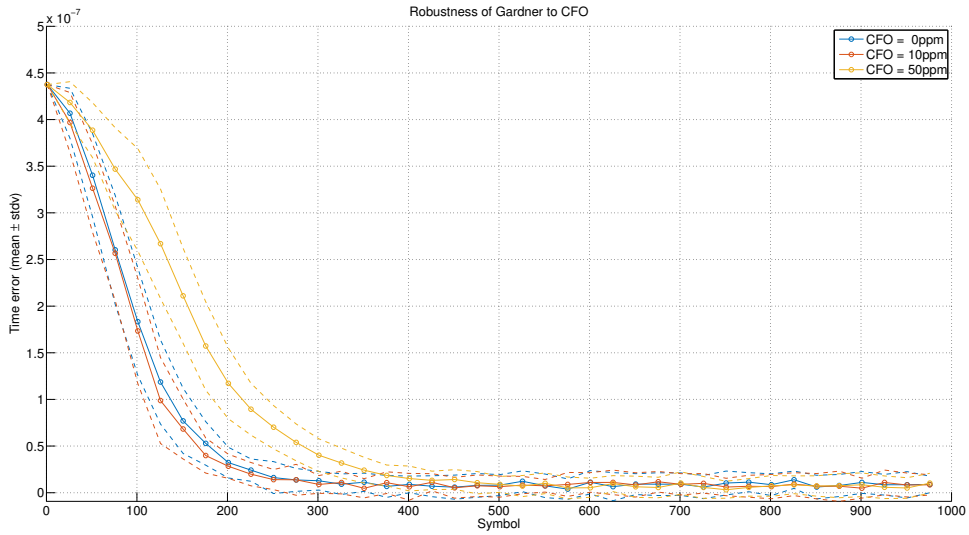

Figure 11: Convergence of the algorithm as a function of the CFO.

that for a standard specified CFO value of 10 ppm, the Gardner algorithm still converges satisfyingly.

2.5 Time of Arrival and CFO Estimation Using Frame Acquisition

In a practical application, the receiver doesn't know when the emitter will start to transmit a message. In the case of a continuous stream, the receiver still has to resynchronize regularly because of the synchronization imperfections we discussed in the introduction to this section. This is done by using differential cross-correlation to a pilot sequence which is repeatedly inserted in the stream of data blocks. Using this cross correlator, it is possible to obtain an estimation of the time of arrival (ToA), meaning the position of the first bit of the frame in the incoming sequence, and of the CFO. In addition to regularly reestimating the CFO and ToA which can slowly drift over time, the regularly spaced pilot sequences allow to refine the CFO estimation using frame interpolation.

The first output metric of the differential cross-correlator is shown in figure 12 in order to illustrate the maximization over n of $\sum_{k=1}^{K} |D_k[n]|$, for $N = 40$ and $K = 8$. As expected, the output metric significantly at $n = 1$, which in this particular case is the true time of arrival.

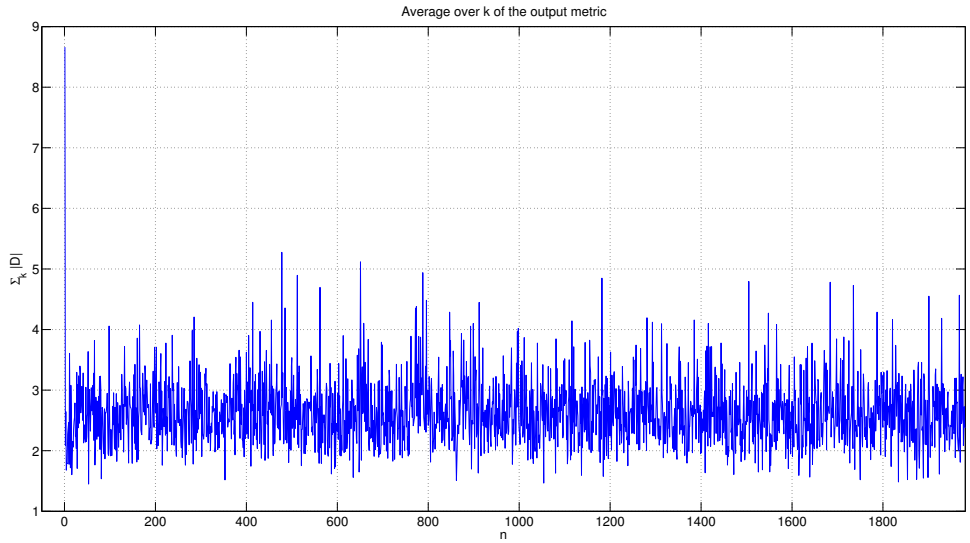

Figure 12: ToA output metric.

Figure 13 shows the accuracy of the ToA estimation in function of N, K and the channel SNR for 4-QAM modulation. Figure 14 shows the resistance of the ToA estimation to CFO for 4-QAM modulation with $N = 40$ and $K = 8$.

Figure 15 shows the accuracy of the CFO estimation as a function of N, K and the channel SNR for 4-QAM modulation. Figure 16 shows the resistance of the CFO estimation to CFO for 4-QAM modulation with $N = 40$ and $K = 8$.

This shows that the ToA estimation is sufficient at realistic noise levels and standard CFO of 10 ppm, but that frame interpolation is indeed needed to be able to correctly demodulate the incoming signal. This is demonstrated in figure 17, which shows that the remaining CFO still introduces a significant phase drift over the course of a single 512 bits block. However, frame acquisition is still able to compensate most of the synchronization error.

2.6 Questions

2.6.1 Simulation

Derive analytically the baseband model of the channel including the synchronisation errors. On figure 18, let us now assume that the

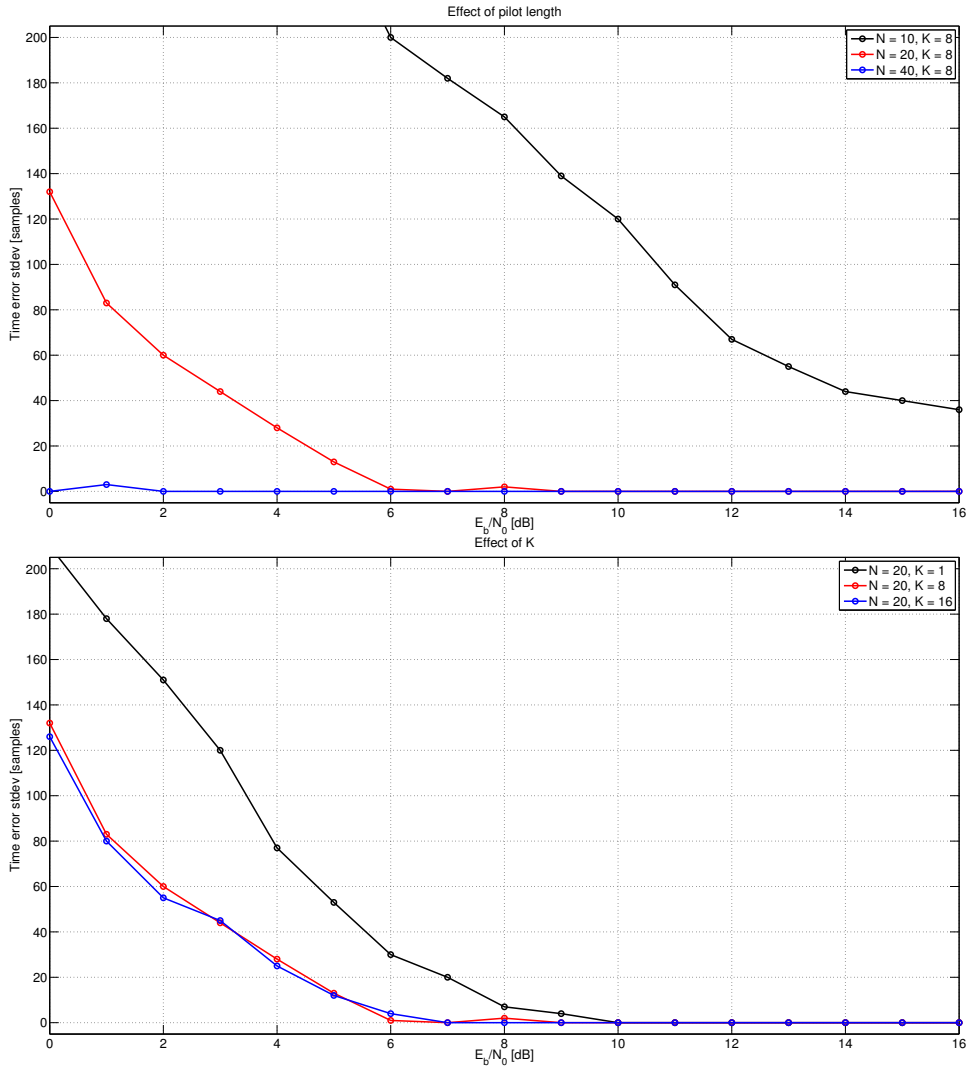

Figure 13: Pilot ToA error as a function of N and K.

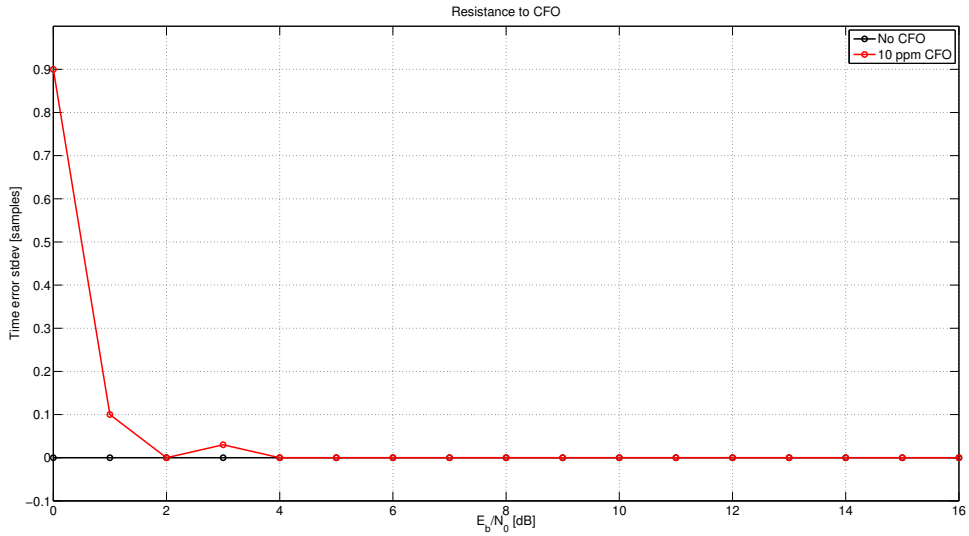

Figure 14: Resistance of the ToA estimation to CFO.

receiver's oscillator differs from the emitter's oscillator by $\Delta\omega$ in pulsation and by φ in initial phase, and let us not consider the additive noise, which has already been covered previously. We can write:

$$\begin{aligned} r(t) = s(t) &= \mathrm{Re}\{e_s(t)\}\cos(\omega_c t) + \mathrm{Im}\{e_s(t)\}\sin(\omega_c t) \\ e_r(t) &= \mathrm{Re}\{e_s(t)\}\left[\cos(\omega_c t)\cos((\omega_c+\Delta\omega)t+\varphi) + \cos(\omega_c t)\sin((\omega_c+\Delta\omega)t+\varphi)\right] \\ &\quad + j\,\mathrm{Im}\{e_s(t)\}\left[\sin(\omega_c t)\cos((\omega_c+\Delta\omega)t+\varphi) + \sin(\omega_c t)\sin((\omega_c+\Delta\omega)t+\varphi)\right] \end{aligned}$$

By using simpson rules, and by removing the high frequency components which are eliminated by the low pass filters, we find:

$$e_r(t) = \mathrm{Re}\{e_s(t)\}\cos(\Delta\omega t+\varphi) - j\,\mathrm{Im}\{e_s(t)\}\sin(\Delta\omega t+\varphi)$$

To obtain the baseband model of the channel, including CFO, we have to find $Z_{cfo}(t)$ such that:

$$\begin{aligned} \mathrm{Re}\left\{Z_{cfo}(t)\cdot e^{j\omega_c t}\right\} &= \mathrm{Re}\{e_r(t)\}\cos(\omega_c t) - j\,\mathrm{Im}\{e_r(t)\}\sin(\omega_c t) \\ \Rightarrow Z_{cfo}(t) &= e_s(t)\cdot e^{j\Delta\omega t+\varphi} \end{aligned}$$

This means that the modulation/demodulation synchronization errors can be modeled in the baseband by multiplying the baseband signal by $e^{j\Delta\omega t+\varphi}$.

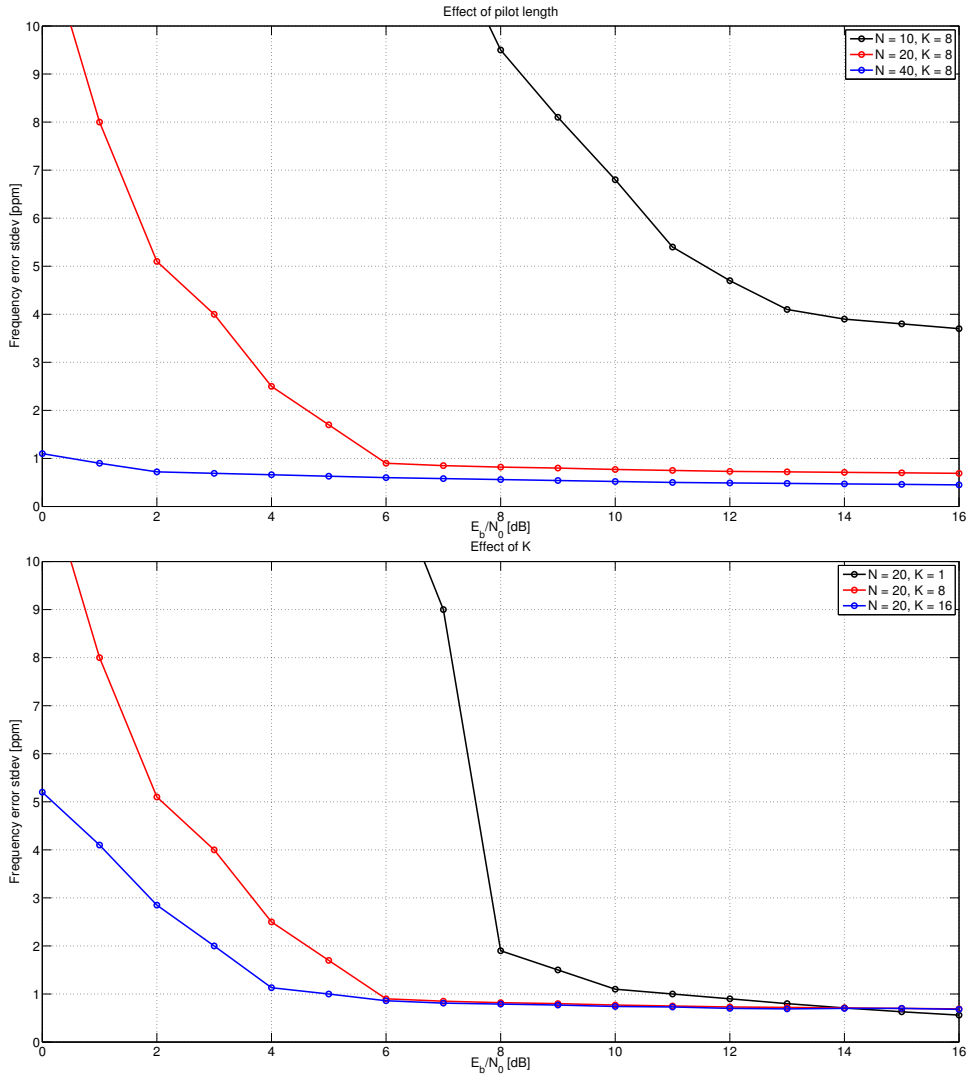

Figure 15: CFO error as a function of N and K.

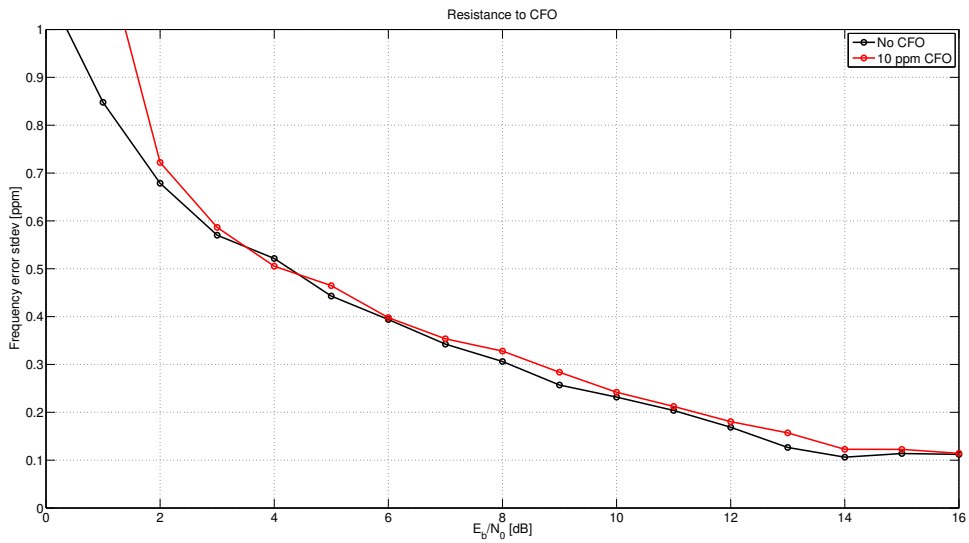

Figure 16: Resistance of the CFO estimation to CFO.

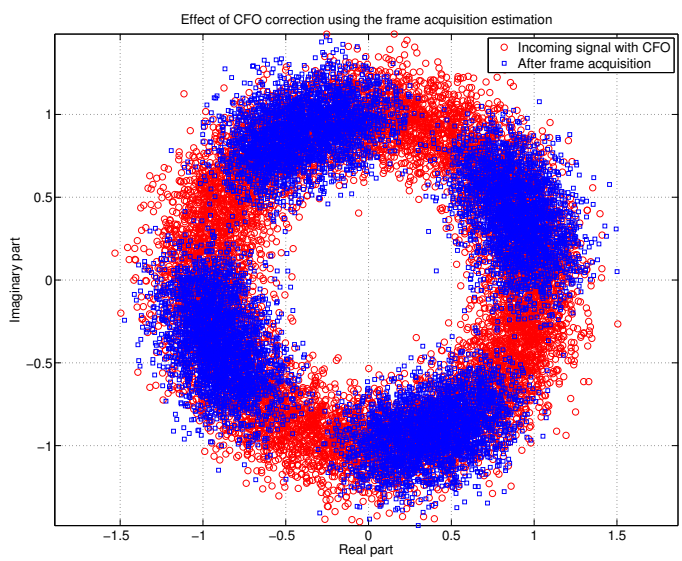

Figure 17: Effect on the constellation of the frame acquisition (25 independent 512 bits blocks, remaining CFO = 2.3 ppm).

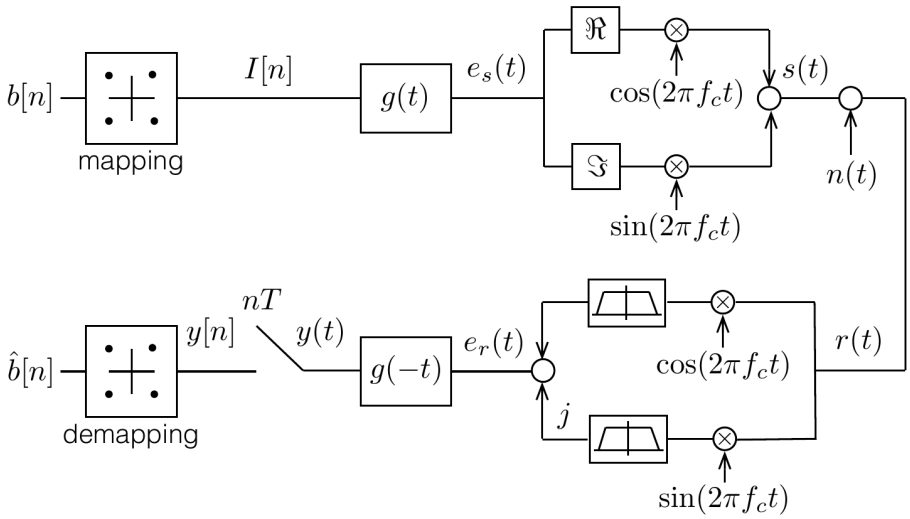

Figure 18: Transceiver block diagram. [Source: course notes]

How do you separate the impact of the carrier phase drift and ISI due to the CFO in your simulation? To only measure the impact of the ISI introduced by the CFO, the CFO is perfectly corrected by hand after the filtering operation at the receiver.

How do you simulate the sampling time shift in practice? The modulated message at the symbol frequency is first upsampled by a ratio $r_{os} = \frac{f_s}{f_m}$ before all the other operations. At the receiver, after the filtering operation, the signal is downsampled. During this downsampling operation, a fixed sampling time shift $t_0 = n \cdot T_s$ ($-\frac{r_{os}}{2} < n \leq \frac{r_{os}}{2}$, n integer) can be introduced by shifting the indexes.

How do you select the simulated E_b/N_0 ratio? E_b/N_0 should be small enough so that every block in the synchronization correction chain can converge correctly. Else, if a block fails, for example Gardner compensation of the sample time shift, then every subsequent block will most likely fail as well, which means that the whole experience is useless. However, E_b/N_0 should still be high enough so that it is realistic and it is possible to observe meaningful error rates. A typical value is 10 dB.

How do you select the lengths of the pilot and data sequences? The pilot's length should be long enough to get a good estimation of the CFO and the ToA. The length of a data sequence is selected to ensure a correct phase interpolation between two pilot sequences. Furthermore, the pilot sequence length and repetition should be as small as those two considerations allow in order to maximize the channel throughput.

2.6.2 Communication System

In which order are the synchronisation effects estimated and compensated. Why? First the sampling time shift is estimated and compensated with Gardner's algorithm, because frame and frequency acquisition can only work on a correctly sampled sequence, while Gardner's algorithm is robust to CFO.

Explain intuitively how the error is computed in the Gardner algorithm. Why is the Gardner algorithm robust to CFO? At step n, the time shift error is estimated by looking at the value of the signal between samples n and n-1. If there was a zero crossing, then the middle value should be zero or very small. The estimation of epsilon is then corrected by a term that is proportional to the value of this middle sample. The "direction" of the correction is determined by the sign of the middle value and the direction of the zero crossing (upwards and negative value means the sampling is "early", upwards and positive value means the sampling is "late"; and vice-versa for downwards crossings).

Explain intuitively why the differential cross-correlator is better suited than the usual cross-correlator? Isn't it interesting to start the summation at $k = 0$ (no time shift)? The differential cross-correlator first estimates the start time, and then the CFO. In doing so, it avoids an exhaustive 2D search which is computationally expensive. The alternative, a simple ToA ML estimation searches over a single parameter, but isn't robust to CFO. It is not interesting to start the summation at $k = 0$ because the term $D_0[n]$ is given by:

$$D_0[n] = \frac{1}{N} \cdot 1 \cdot \sum_{l=0}^{N-1} \left[\left| I[n] \right|^2 \left| a[l] \right|^2 \right]$$

This shows that it only depends on the average power of the window, and so it couldn't provide additional information.

Are the frame and frequency acquisition algorithms optimal? If yes, give the optimisation criterion. The differential cross correlator algorithm isn't optimal because it doesn't optimize over n and $\Delta\omega$ at once. However, it is sufficient for practical applications with CFO $\lesssim 10\,\text{ppm}$ and reasonable amounts of noise $\frac{E_b}{N_0} \gtrsim 4\,\text{dB}$

3 Low-density parity check code

To improve the reliability of the channel, Low Density Parity Check (LDPC) is implemented in this section. Its aim is to add redundancy to the information sent by the transmitter and use these additional bits to detect and correct errors. LDPC codes have the property that the generator and check matrices are sparse, which allows a lower complexity implementation.

3.1 LDPC encoding and hard decoding

Figure 19 illustrates the channel coding gain as a function of the number of iterations. Hereafter, the information bits are coded using (256,128) LDPC.

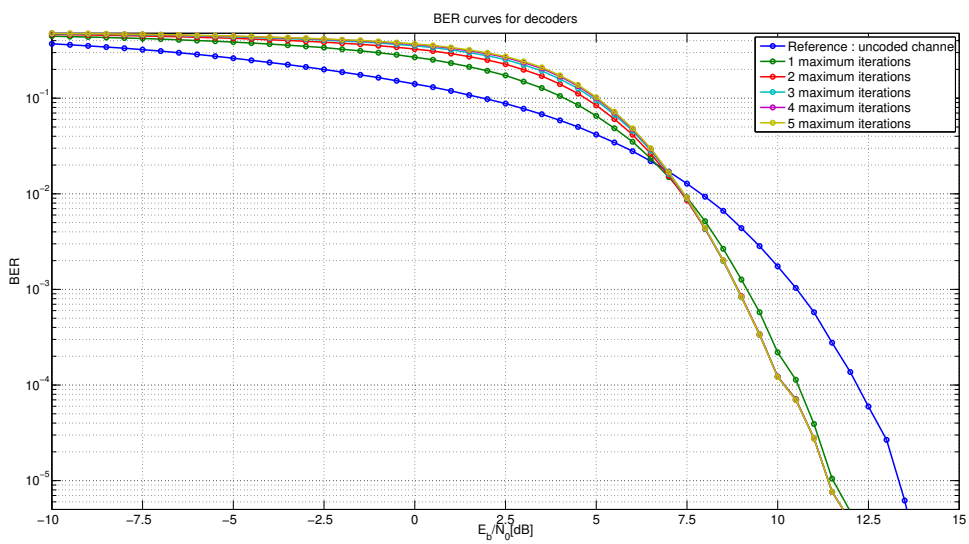

Figure 19: BER curves for different numbers of hard decoding iterations, compared with previous results.

3.2 Soft decoding

Figure 20 compares the BER curves for the channel without coding, with coding and hard decoding and with different number of soft decoding iterations. As will be explained in the next subsection, BPSK is used for this

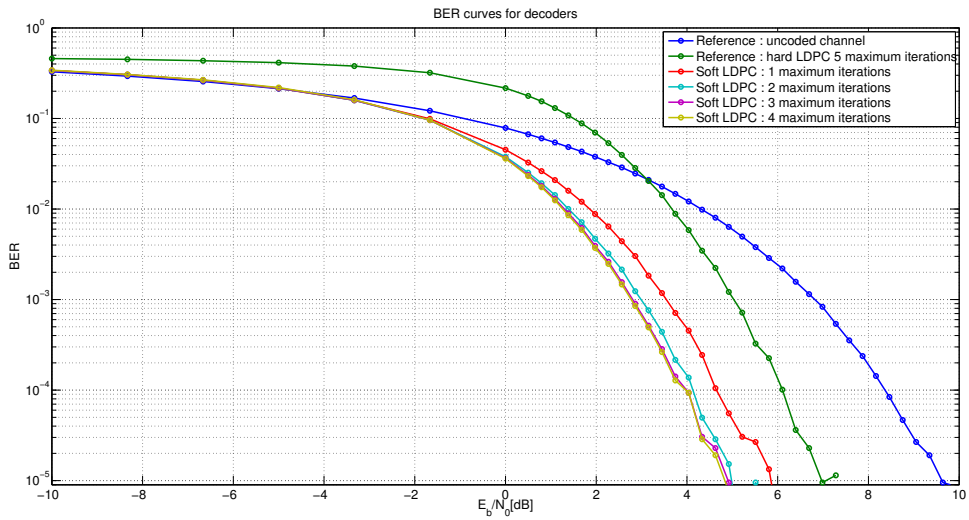

Figure 20: BER curves for different numbers of soft decoding iterations, compared with hard decoding and no coding.

experiment. We observe that, as expected, the soft decoder achieves better coding gain the the hard decoder. Moreover, it doesn't introduce additional errors at lower SNR. This is also due to how the decoder decides to stop iterating, since it stops iterating as soon as the number of detected errors increases.

3.3 Questions

3.3.1 Simulation

When building the new BER curves, do you consider the uncoded or coded bit energy on the x-axis? Here we consider the coded bit energy on the x-axis, but both options are interesting. In a low-power application, the relevant metric is energy spent per sent bit of information in order to achieve a given error rate, which means the uncoded bit energy should be used on the x-axis. If the emitter has constant power, the relevant metric is the coded bit energy which is defined as $\frac{P}{f_m \cdot \log_2(N_{sym})}$.

How do you limit the number of decoder iterations For hard decoding, the decoding stops as soon as one decoding iteration does not update the coded block, or when the maximum number of iterations is reached. This maximum number of iteration is set to 5. Figure 19 shows that very few additional errors are corrected past this value. For soft decoding, the decoding stops as soon as there are no more detected errors, or when an iteration has actually introduced errors, or when the maximum number of iterations is reached.

Why is it much simpler to implement the soft decoder for BPSK or QPSK than for 16-QAM or 64-QAM? For BPSK and QPSK, there is no need to compare euclidean distances to the different points of the constellation. For BPSK, the probability that the received bit is a one or zero is simply given by the real part of the received symbol. For QPSK, the probability that the first bit is a one or zero is given by the real part of the received symbol, and the probability that the second bit is a one or a zero is given by its imaginary part.

3.3.2 Communication system

Demonstrate analytically that the parity check matrix is easily deduced from the generator matrix when the code is systematic. A systematic code makes the information bits appear as is in the latter part of the coded block. Thus, the generator matrix can be written as $G = [PI]$. The parity check matrix is orthogonal to the generator matrix, so it must verify $G \cdot H^T = \overline{0}$. The solution to this identity is simply $H = [IP^T]$. Indeed:

$$[PI] \cdot [IP^T]^T = P \oplus P = \overline{0}$$

Explain why we can apply linear combinations on the rows of the parity check matrix to produce an equivalent systematic code. The important property of the parity check matrix is that it spans the vector subspace complementary to the codewords subspace spanned by the generator matrix. Linear combinations of the base vectors that form the columns of H still span the same subspace, so it produces an equivalent code.

Why is it especially important to have a sparse parity check matrix (even more important than having a sparse generator matrix)? Sparsity allows computational optimizations at various levels, but

more specifically, it tremendously reduces the number of messages and responses between variable nodes and check nodes. This is because in the case of a sparse parity check matrix:

- Each check node operates on a low number of variable nodes (few ones in every row).
- Each variable node intervenes in a low number of variable nodes (few ones in every column).

Explain why the check nodes only use the information received from the other variable nodes when they reply to a variable node. In both hard- and soft-decoding, the messages exchanged between c-nodes and v-nodes must be stochastically independent. This means that when a node responds to another (regardless of the direction), it must only use extrinsic information, meaning that it shouldn't use the information from the node it is responding to.

List of Figures

1 Block diagram of the communication system. 2
2 Nyquist filtering limits the communication bandwidth. 3
3 Cancellation of the inter symbol interference of a raised cosine filter. 3
4 BER in function of $\frac{E_b}{N_0}$ for different QAM modulations. 4
5 Baseband model with synchronization errors. 7
6 Impact of the ISI introduced by the CFO on the BER. 8
7 Effect of 10 ppm of CFO on the ISI of the pair of filters. . . . 9
8 Constellation at different stages in the communication. 9
9 Impact of the sample time shift on the error rate. 10
10 Convergence of the algorithm as a function of κ. 10
11 Convergence of the algorithm as a function of the CFO. . . . 11
12 ToA output metric. 12
13 Pilot ToA error as a function of N and K. 13
14 Resistance of the ToA estimation to CFO. 14
15 CFO error as a function of N and K. 15
16 Resistance of the CFO estimation to CFO. 16
17 Effect on the constellation of the frame acquisition 16
18 Transceiver block diagram. 17
19 BER curves for different numbers of hard decoding iterations, compared with previous results. 19
20 BER curves for different numbers of soft decoding iterations, compared with hard decoding and no coding. 20