Brussels Faculty of Engineering

Modulation & Coding

ELEC–H401

Simulation of a DVB–S2 Communication Chain

Authors:

Cédric Hannotier

Badr-Ali Mouaden

Mahtieu Petitjean

June 11, 2018

Contents

1 Introduction

This report presents the simulation of a DVB-S2 communication chain. The most important design parameters were varied in order to illustrate their impact on the quality of the communication via the bit error rate (BER).

This project was divided in three steps. Each of these steps is presented, then various graphs illustrate the important concepts and results. To conclude each section, questions regarding the simulation and the communication chain itself are answered.

2 Optimal communication chain over the ideal channel

2.1 Simulation

The implementation scheme is shown in Figure 1.
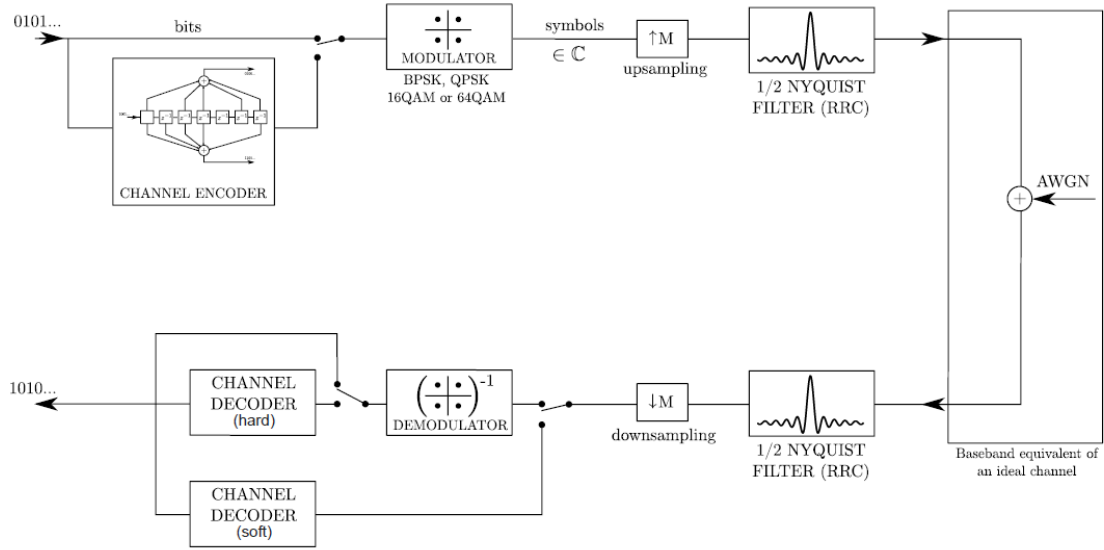
Figure 1: Block scheme of the ideal channel

The various components of the communication chain are detailed hereafter. The channel encoder is not yet implemented, and the symbols mapping/demapping were provided.

2.1.1 Halfroot Nyquist filter

After the mapping, the complex symbols are convolved with a root raised-cosine filter. It allows to limit the communication bandwidth, as illustrated in Figure 2 where the PSD of the symbol stream are shown before and after filtering, with a rolloff factor β of 0.3. The symbol frequency was fixed to 2 MHz, which translates to a cutoff frequency of 1 MHz.
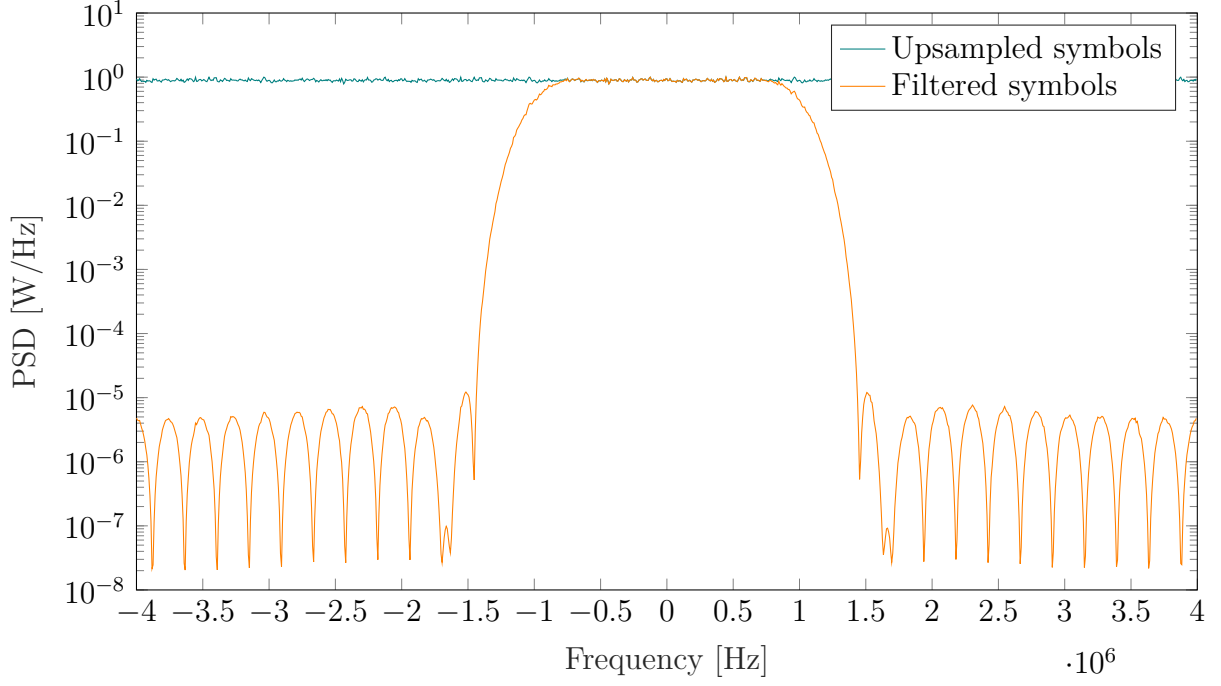
Figure 2: PSD of the symbol stream at the transmitter before and after filtering

At the receiver side, the stream is convolved again with the matched filter, forming a Nyquist filter and maximizing the SNR. The Nyquist filter allows to cancel the ISI in the sense that it reduces to a Dirac pulse when sampled at the symbol frequency, as shown in Figure 3. Hence, no symbol of the symbol stream overlaps with the current received symbol.

Figure 3: Illustration of the cancellation of ISI by the Nyquist halfroot filter

2.1.2 Noise addition

The channel is corrupted by Additive White Gaussian Noise (AWGN), which is simulated using its baseband equivalent representation. The channel was simulated for various SNR (defined w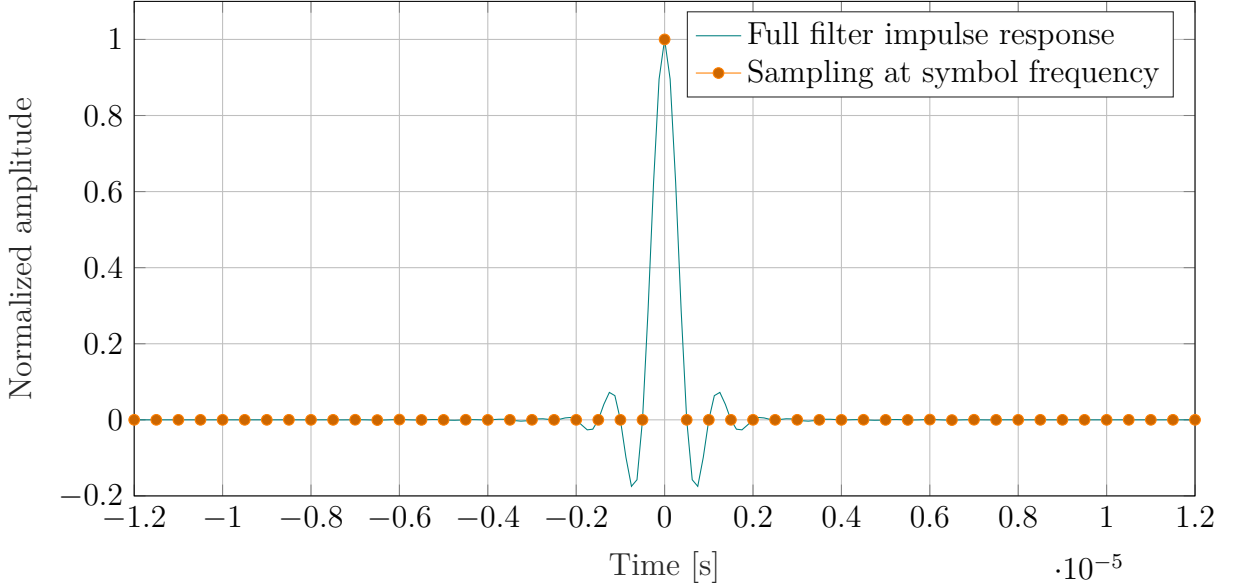ith respect to the bit energy) and various modulation schemes, and the results are plotted in Figure 4. The simulations confirm the theoretical results.
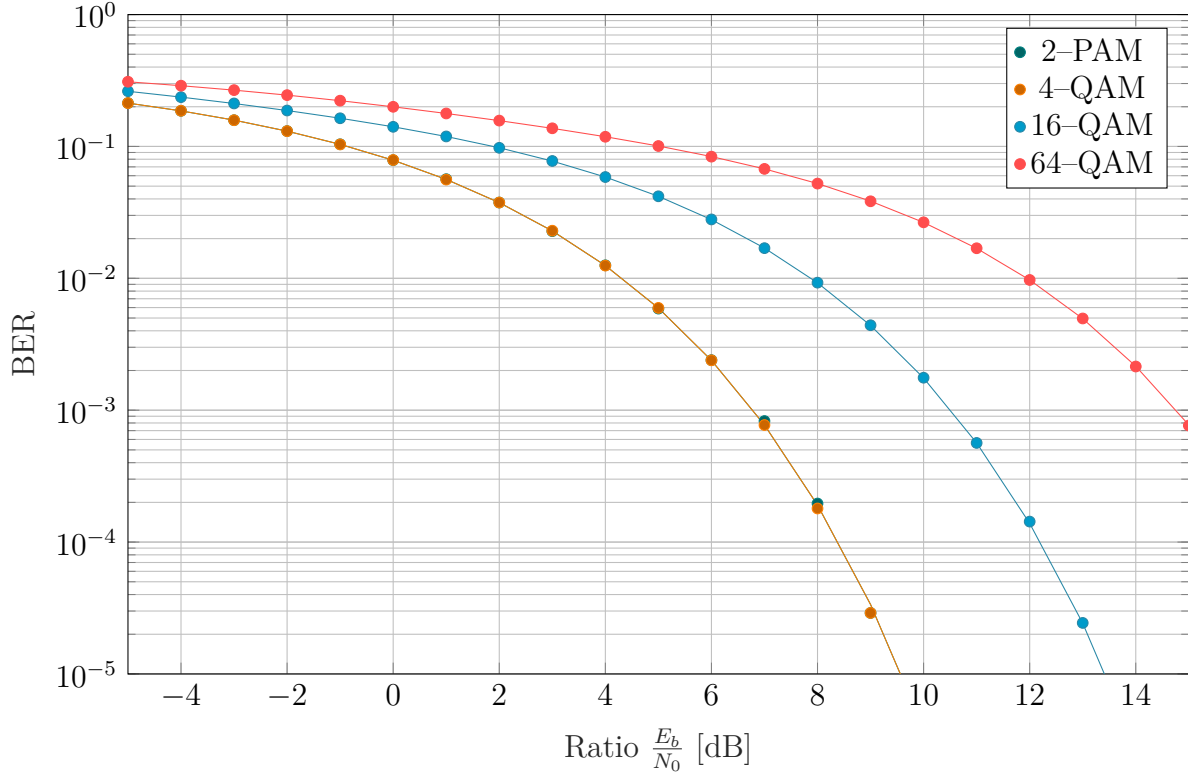
Figure 4: Bit error rate as a function of the SNR for various modulations. The full lines correspond to the theoretical curves and the dots to the simulation.

As expected, more energy is required to have the same BER when the constellation size is increased. It is worth noting that the worst BER value that could be achieved is 0.5, corresponding to a random choice of the binary values. Moreover, the BER curves for 2-PAM and 4-QAM are superposed because a 4-QAM modulation can be seen as two 2-PAM modulations in quadrature.

2.2 Questions

2.2.1 Questions regarding the 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 ?

The baseband model allows to implement the simulations regardless of the carrier frequency. If the system were simulated in bandpass, the required sampling frequency would be extremely high and unrealistic.

How do you choose the sample rate in Matlab ?

The sampling rate should be at least twice as high as the symbol frequency. It will be increased further when simulating the sample time shift in the third step.

How do you make sure you simulated the desired E_b/N_0 ratio ?

First the energy of a bit is evaluated, then the noise power is computed in order to obtain the desired E_b/N_0 ratio. The energy of the bit E_b is given by

$$E_b = \frac{T_{samp}}{2N_{bits}} \sum_n |s\,[n]|^2$$

Then N_0 can be chosen according to the wanted SNR and the noise is added to the symbols using the baseband representation :

$$r\,[k] = \sqrt{N_0 f_{samp}} \left\{ \mathcal{N}(0,1) + j\mathcal{N}(0,1) \right\}$$

The square root is present because of the baseband representation of the noise. Indeed, the computed power is equally distributed between the real and imaginary parts of the noise.

How do you choose the number of transmitted data packets and their length ?

A rule of thumb is to send 10^{n+2} bits of data in order to reliably observe of BER of 10^{-n} as it corresponds to 100 detected errors. Moreover, the total number of bits should be a multiple of the number of bits per symbol used in the modulation scheme. The Figure 4 was realised by sending around 10^8 bits of data.

2.2.2 Questions regarding the communication system

Determine the supported (uncoded) bit rate as a function of the physical bandwidth.

The bit rate R is given by $R = \frac{log_2(M)}{T}$ where M is the number of symbols and T is the symbol duration. The Nyquist filtering yields a relationship between the symbol duration and the bandwidth, $\Delta f = 2/T$ (this can be seen on Figure 2 where T = 0.5 µs and $\Delta f = 2$ MHz). This gives:

$$R = \frac{log_2(M) \ \Delta f}{2}$$

Explain the trade-off communication capacity/reliability achieved by varying the constellation size.

As detailed in the previous question, increasing the constellation size (the number of bits transmitted per symbol M) allows to increase the bit rate. However, the minimum euclidean distance between the symbols in the constellation size decreases, which causes the error probability to increase. As a result, to achieve a similar BER, more energy is required when the constellation size increases but the bit rate also increases.

Why do we choose the halfroot Nyquist filter to shape the complex symbols ?

The halfroot Nyquist filter allows to limit the communication bandwidth (it is more spectrally efficient than the plain rectangular window) and allows to cancel ISI.

How do we implement the optimal demodulator? Give the optimization criterion.

The optimal demodulator is implemented as a bank of K filters matched to the basis functions of the modulation scheme:

$$h_i(t) := s_i(-t) \qquad i = 1, ..., K$$

The outputs to such filters are the autocorrelations between the input signal and the basis functions. The filters are mathematically equivalent to a bank of K correlators. The optimisation criterion is the maximisation of the SNR at the output. It is realized by the matched filters, and the SNR will be equal to $\frac{2\mathcal{E}}{N_0}$.

How do we implement the optimal detector? Give the optimization criterion.

There exist two possible optimization criteria for the detector. First, the maximum a posteriori (MAP) criterion, which minimises the error probability by exploiting the knowledge of the a priori probabilities $p(\underline{s}_m)$ and the conditional probabilities $p(\underline{r}|\underline{s}_m)$, where $\underline{r}$ is the observation. It selects the symbol which is the most likely to have been sent, knowing that we observed $\underline{r}$.

$$\tilde{\underline{s}}_m^{\mathrm{MAP}} = \max_{\underline{s}_m} p(\underline{s}_m|\underline{r}) \overset{\mathrm{Bayes}}{=} \max_{\underline{s}_m} p(\underline{r}|\underline{s}_m)p(\underline{s}_m)$$

The maximum likelihood criterion is equivalent to the MAP criterion when all the M symbols are equally probable, so that $p(\underline{s}_m) = 1/M$ is constant and can be dropped when maximizing the probabilities.

$$\tilde{\underline{s}}_m^{\mathrm{ML}} = \max_{\underline{s}_m} p(\underline{r}|\underline{s}_m)$$

The ML criterion determines what symbol is the most likely to have produced the received signal by selecting the symbol which has the lowest euclidean distance to the received symbol.

3 Low-density parity check code

3.1 Simulation

The channel encoder that was skipped in section 2 is now implemented as a low-density parity check code (LDPC) encoder. Soft and hard decoding have been implemented with a code rate of 1/2 and their performance is compared next.

The channel coding adds structured redundancy to the information sent by the transmitter to be able to detect and correct errors at the receiver side. The LDPC code has the property that the check matrix is sparse, allowing a lower complexity of the implementation.

3.1.1 Hard decoding

The hard decoding is based on the implementation of a Tanner graph. The influence of the maximal number of iterations on the BER curves is shown in Figure 5 for a 16-QAM modulation. The conclusions can be generalized to any modulation.
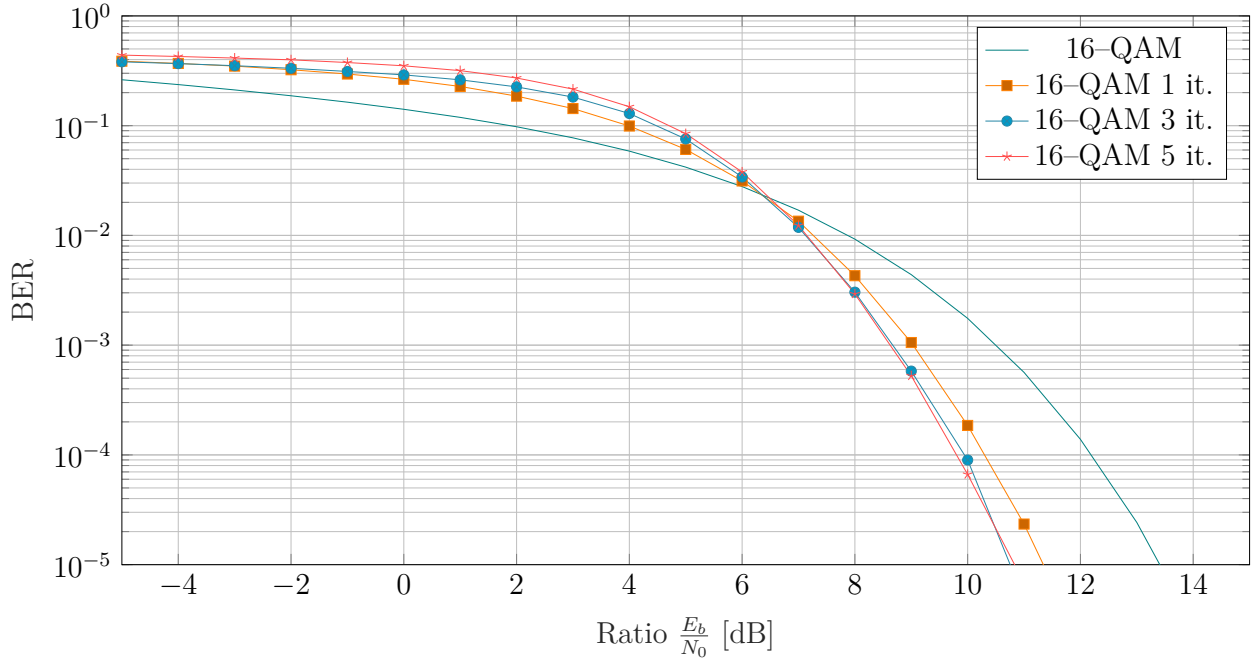

Figure 5: Effect on the BER curves of Hard LDPC decoding for a 16-QAM modulation

For the hard decoding, the performance is worsened at low SNR. Indeed, a noise which is too important causes the code to add errors when it is trying to correct them. Increasing the maximal number of iterations allows a slight improvement in performance.

3.1.2 Soft decoding

In practice, hard decoding is never used as the soft decoding allows for much better performance. Instead on basing the decision on the received bits, it computes probablities based on the received symbols. The results of the simulations for a 2-PAM is shown in Figure 6.
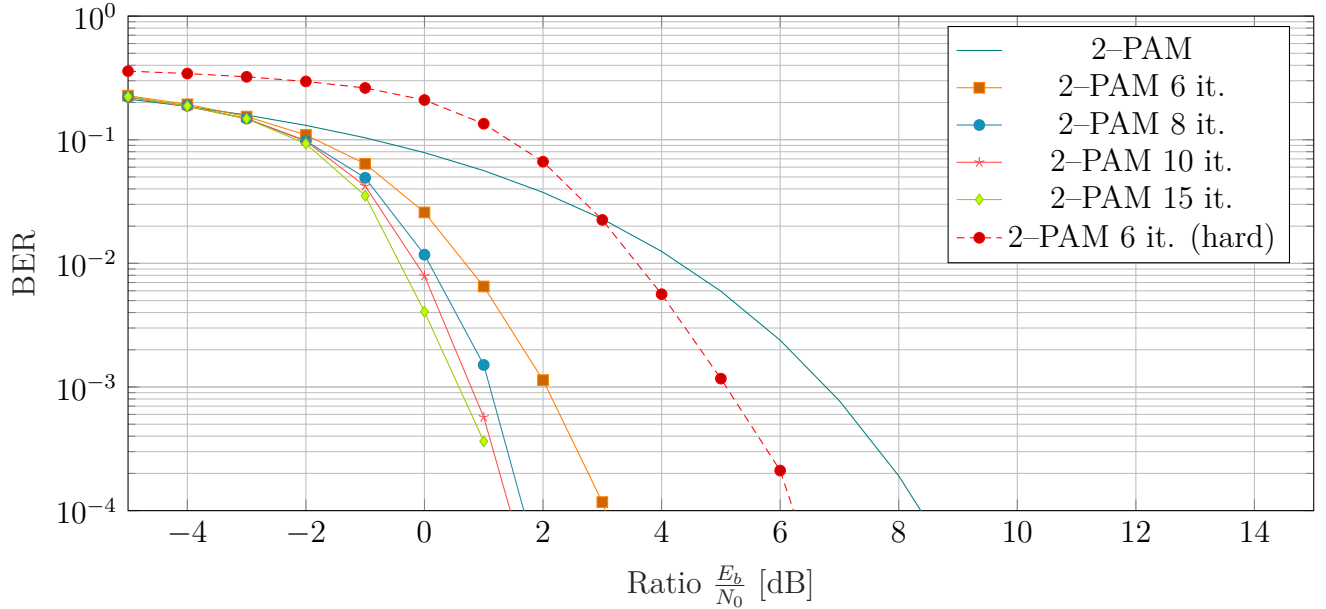

Figure 6: Effect on the BER curves of Hard LDPC decoding for a 16-QAM modulation

The theoretical curve for the ideal communication chain as well as the curve for hard decoding are shown as references. The improvement is significant with soft decoding.

3.2 Questions

3.2.1 Questions regarding the simulation

When building the new BER curves, do you consider the uncoded or coded bit energy on the x-axis?
The coded bit energy is considered here.

How do you limit the number of decoder iterations?
For the hard decoding, the iterations stop either when the maximum number of iterations is reached or when the syndrome is zero.
For the soft decoding, stop either when the maximum number of iterations is reached or when there is no more detected errors.

Why is it much simpler to implement the soft decoder for BPSK or QPSK than for 16-QAM or 64-QAM?

In the case of the 16-QAM or 64-QAM, the euclidean distance should be computed when decoding. This is not the case for BPSK and QPSK, where the decision can be made only by looking at the real and imaginary part of the received symbols.

3.2.2 Questions regarding the communication system

Demonstrate analytically that the parity check matrix is easily deduced from the generator matrix when the code is systematic.

A code is systematic when the mapping is such that part of the code vector coincides with the message vector. In that case, the generator matrix has the form $\underline{\underline{G}} = \left[\underline{\underline{P}}|\underline{\underline{I}}\right]$, with $\underline{\underline{P}}$ the parity array and $\underline{\underline{I}}$ the identity matrix of the corresponding size. The parity check matrix should be created so that its rows are orthogonal to the rows of the generator matrix:

$$\underline{\underline{G}} \cdot \underline{\underline{H}}^T = \underline{\underline{0}}$$

The solution to this equation is given by $\underline{\underline{H}} = \left[\underline{\underline{I}}|\underline{\underline{P}}^T\right]$. Indeed:

$$\underline{\underline{G}} \cdot \underline{\underline{H}}^T = \left[\underline{\underline{P}}|\underline{\underline{I}}\right] \cdot \left[\underline{\underline{I}}|\underline{\underline{P}}^T\right]^T = \underline{\underline{P}} \oplus \underline{\underline{P}} = 0$$

Explain why we can apply linear combinations on the rows of the parity check matrix to produce an equivalent systematic code.

The rows of the parity check matrix are the basis vectors of a subspace which is perpendicular to the subspace spanned by the generator matrix. A linear combination of these basis vectors will still be a basis for the same subspace and the resulting code will be equivalent.

Why is it especially important to have a sparse parity check matrix (even more important than having a sparse generator matrix)?

A sparse parity check matrix allows to significantly reduce the complexity of the decoder. Indeed, element H_{ij} being equal to 1 results in a logical connection in the Tanner graph between the check-node c_i and the variable-node v_j. Hence, reducing the number of 1's in $\underline{\underline{H}}$ reduces the number of exchanged messages and the number of computations.

Explain why the check nodes only use the information received from the other variable nodes when they reply to a variable node.

The information exchanged between the check and variable nodes has to be statistically independent for hard and soft decoding. This means that only the information which is not related to the current node is used to compute the answer.

4 Time and frequency synchronisation

4.1 Simulation

In this section, the synchronisation errors caused by the hardware, the carrier phase and frequency shift are simulated, then algorithms allowing to reduce them are introduced.

4.1.1 Synchronisation errors

On Figure 7 is shown how the CFO affects the constellation. The linearly increasing phase shift caused by the CFO causes a rotation of the symbols. This effect is first manually corrected to study the impact of ISI only.
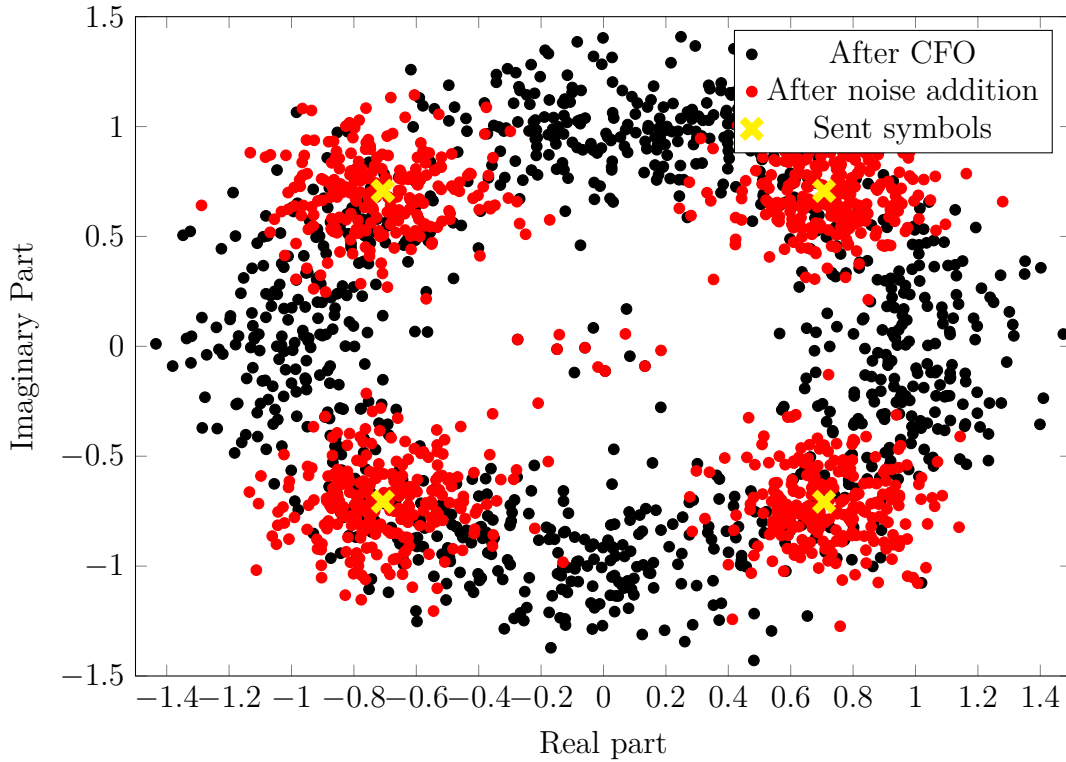

Figure 7: Illustration of the effect of the CFO phase shift on a 4-QAM constellation for $\frac{E_b}{N_0} = 15$dB

The effect of the CFO ISI on the BER curves is shown in Figure 8. Low values of a frequency mismatch between the emitter and receiver quartz (expressed in parts per million, *ppm*) do not disturb the BER too much but increasing value prevent correct communication.
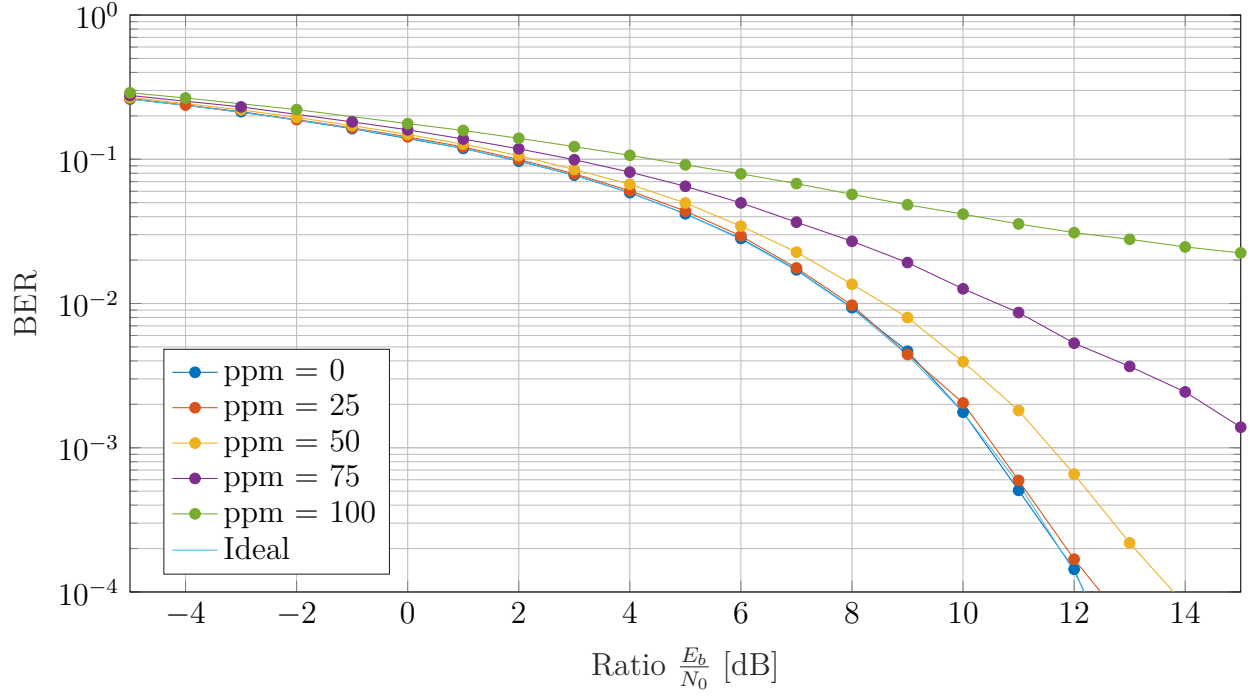
Figure 8: Illustration of the effect of the CFO ISI on the BER curves for a 4-QAM modulation

Next, a sampling time shift is simulated. The impact of this shift is depicted in Figure 9.
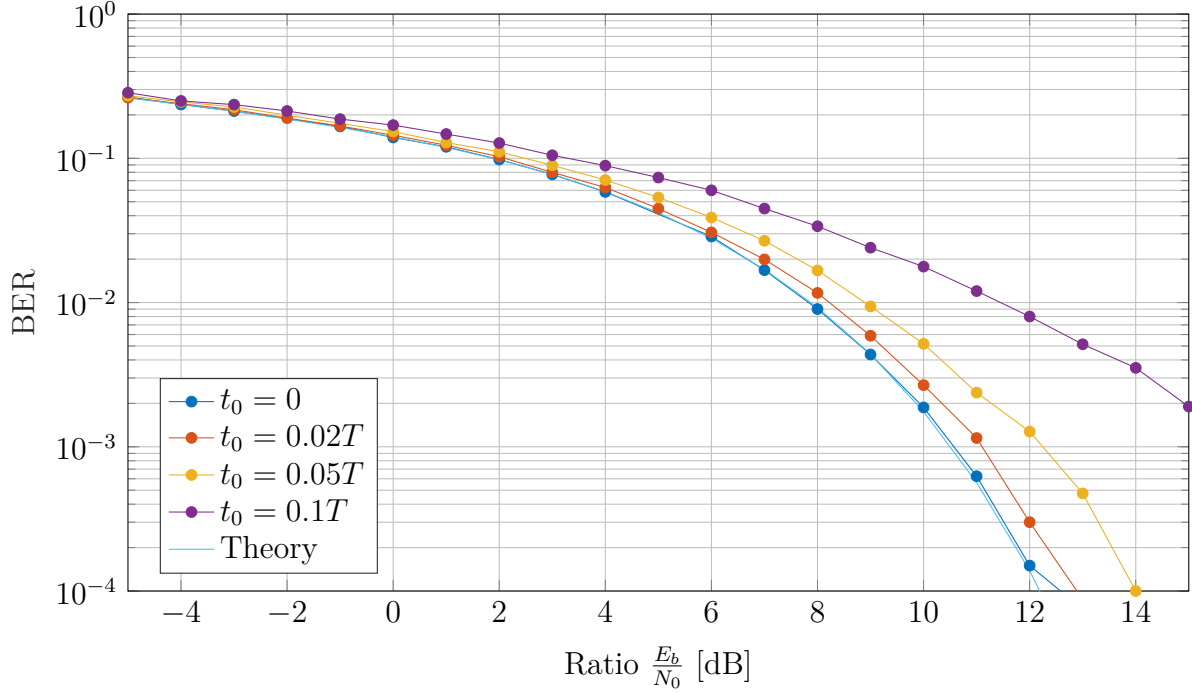

Figure 9: Effect of the sampling shift on the BER for 4-QAM

4.1.2 Correction algorithms

As explained later, the sampling time shift is the first unwanted effect that is tackled, thanks to the Gardner algorithm. Figure 10 shows how the Gardner algorithm is able to correct the sampling time shift.
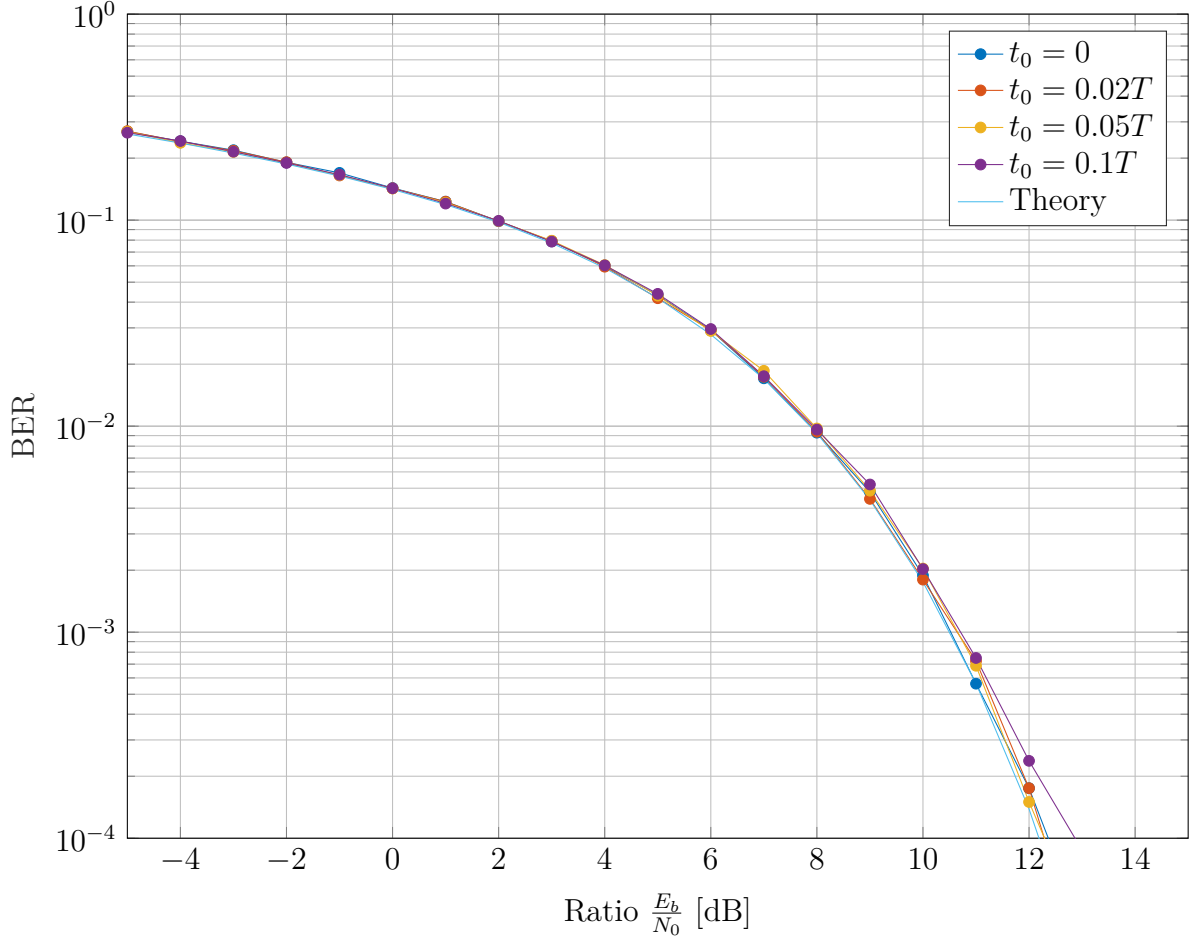

Figure 10: Gardner compensation of the sampling shift without CFO for a 4-QAM modulation

The main parameter of the Gardner algorithm is the factor κ. Figure 11 shows how increasing κ allows to have faster convergence, but at the cost of a less stable estimate. All the subsequent Figures have been generated for a 4-QAM modulation.
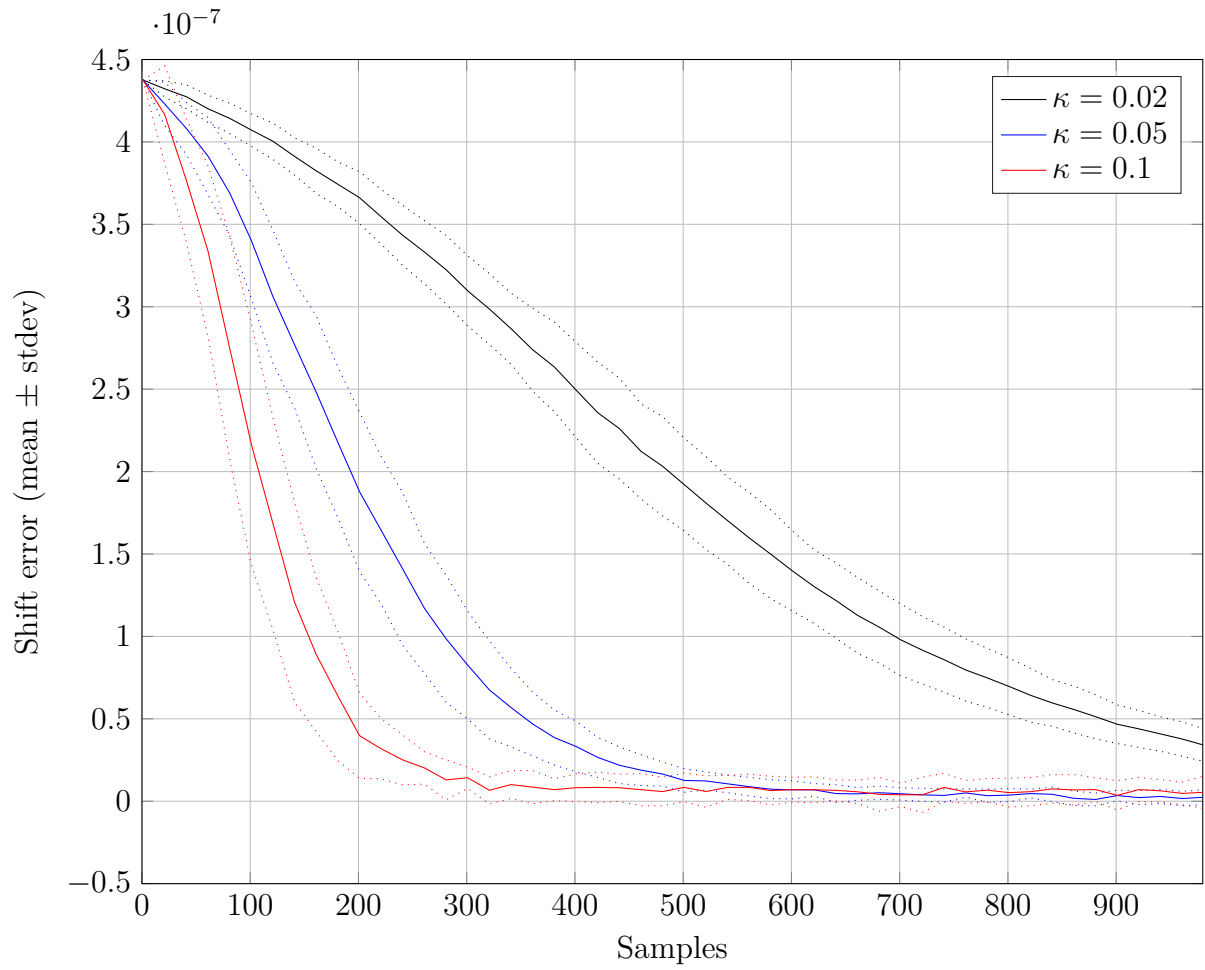
Figure 11: Influence of κ on the convergence of the Gardner algorithm

The Gardner algorithm is robust to CFO, as shown in Figure 12. Its convergence is nearly unaffected by the presence of realistic CFO.
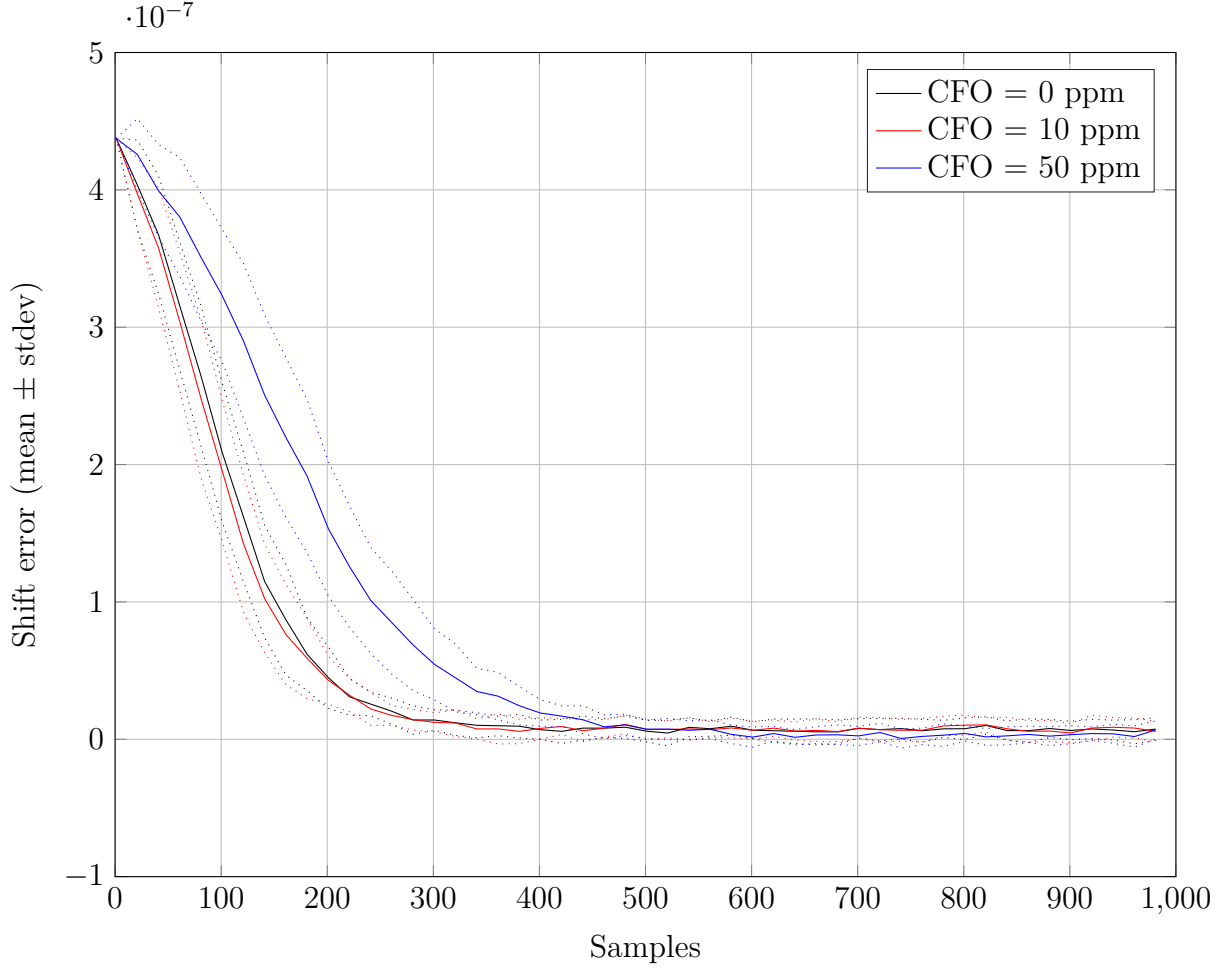
Figure 12: Illustration of the robustness of the Gardner to CFO, $\kappa = 0.1$

After the correction of the sampling time shift, the CFO and the Time of Arrival (ToA) are estimated by sending a pilot (a known sequence) in between frames of data. The main parameters are the length of the pilot N and the size of the averaging windows K. The following Figures depict the depency of the estimation of the ToA and the CFO on these parameters. It appears that when $N = 40$ and $K = 16$, both estimations can be trusted when the SNR is high enough ($> 5dB$).
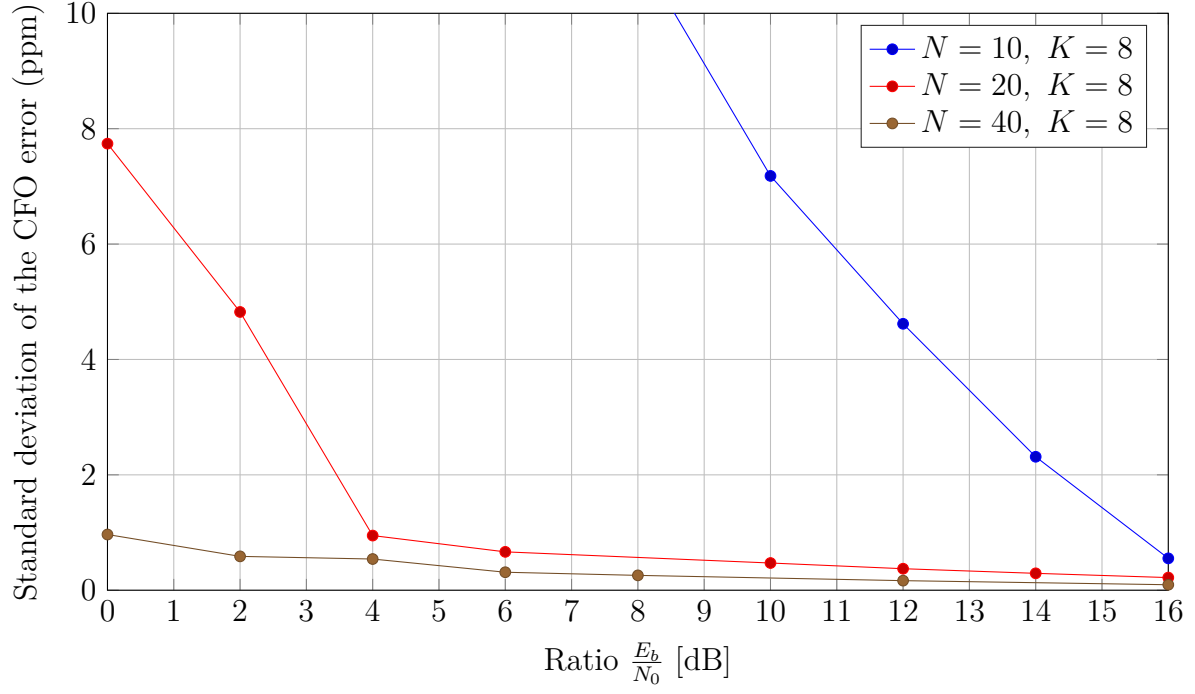
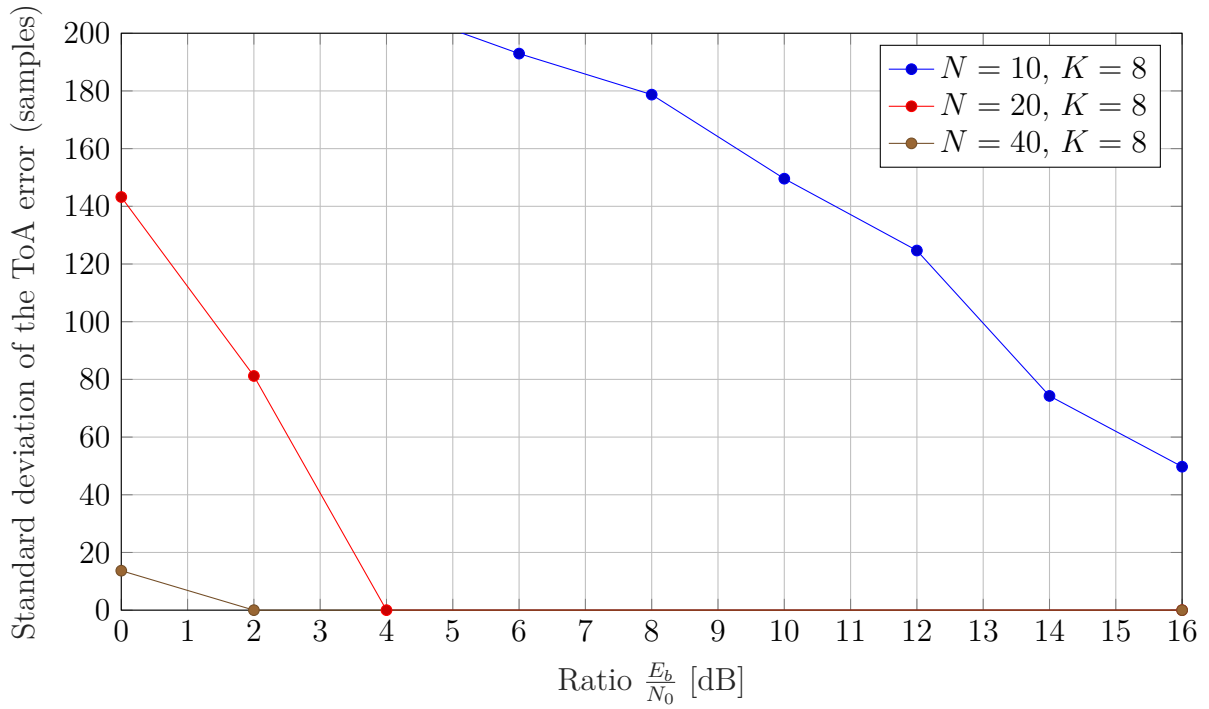Figure 13: Impact of the pilot length on the CFO estimation for a 4-QAM

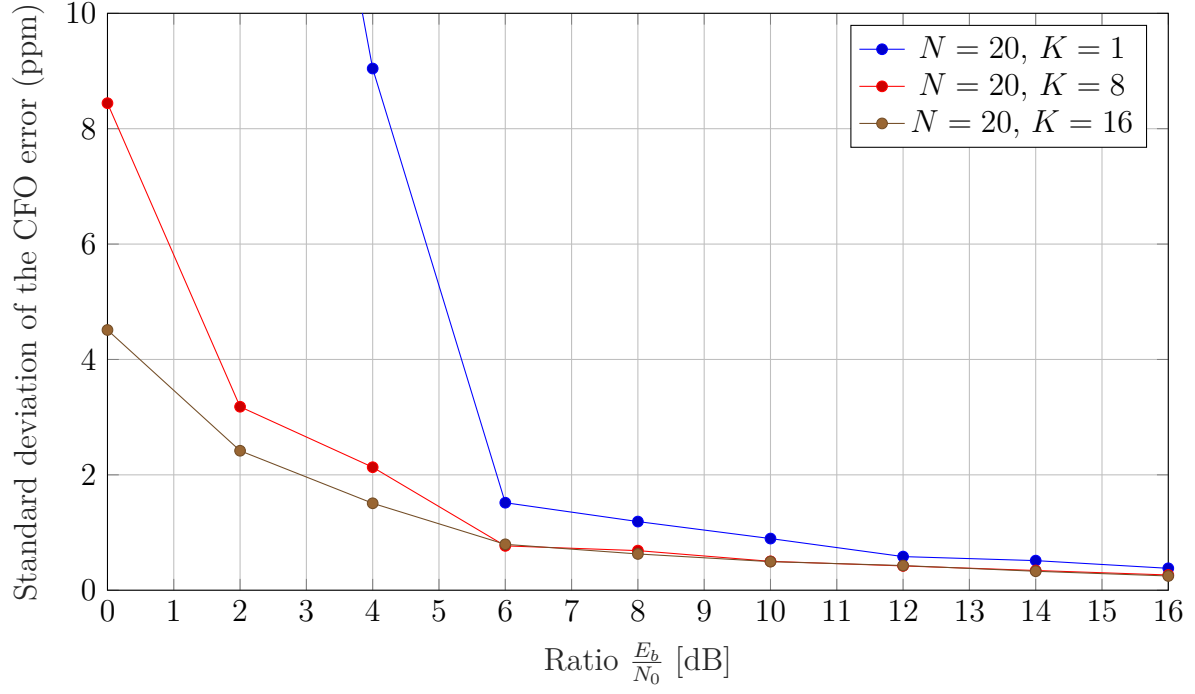Figure 14: Impact of the pilot length on the ToA estimation for a 4-QAM

14

Figure 15: Impact of the size of the averaging window K on the CFO estimation for a 4-QAM
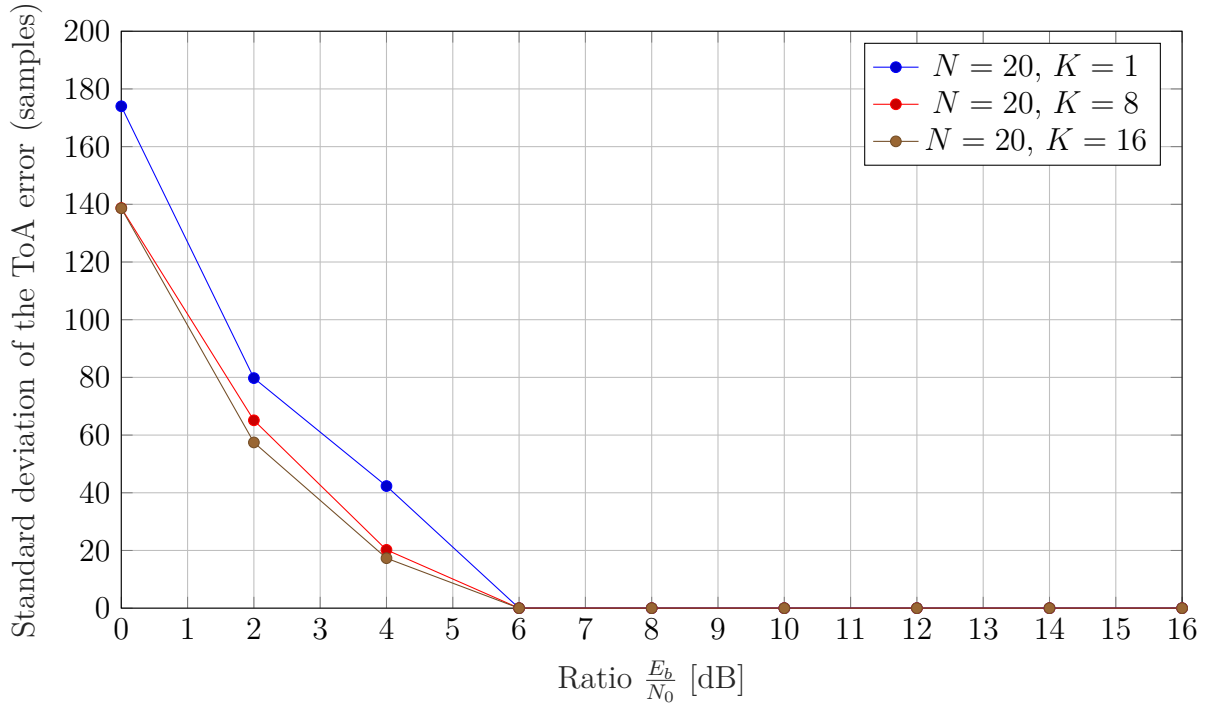

Figure 16: Impact of the size of the averaging window K on the ToA estimation for a 4-QAM

15

On the next two figures, the robustness of the frame acquisition to CFO is clearly visible as it has nearly no impact on the estimations.
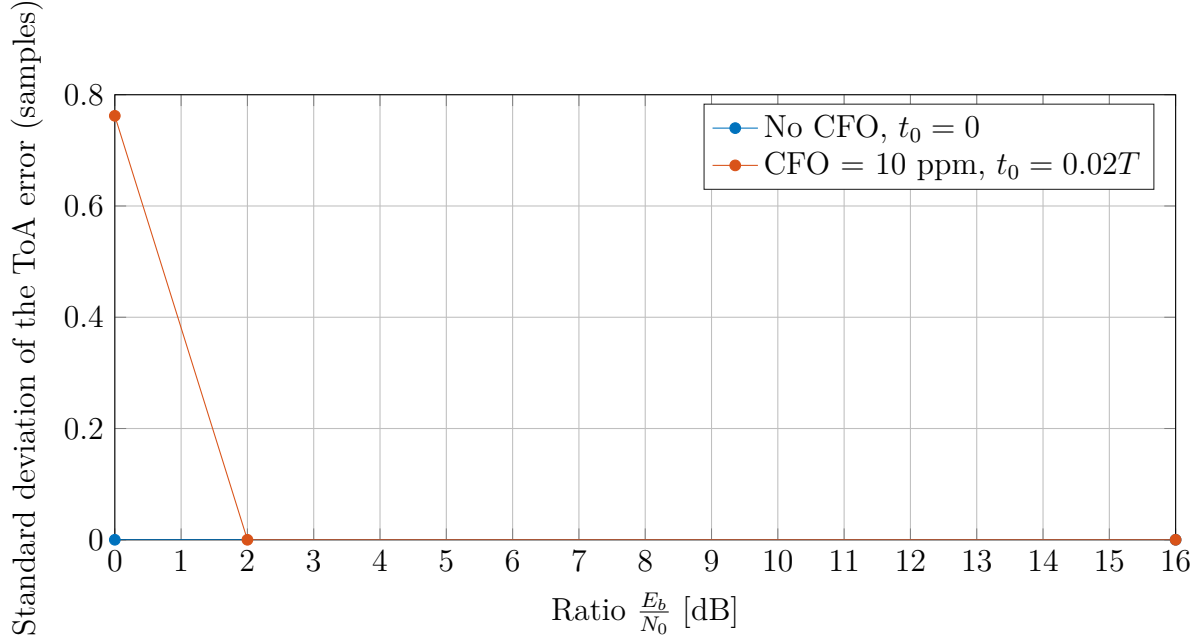

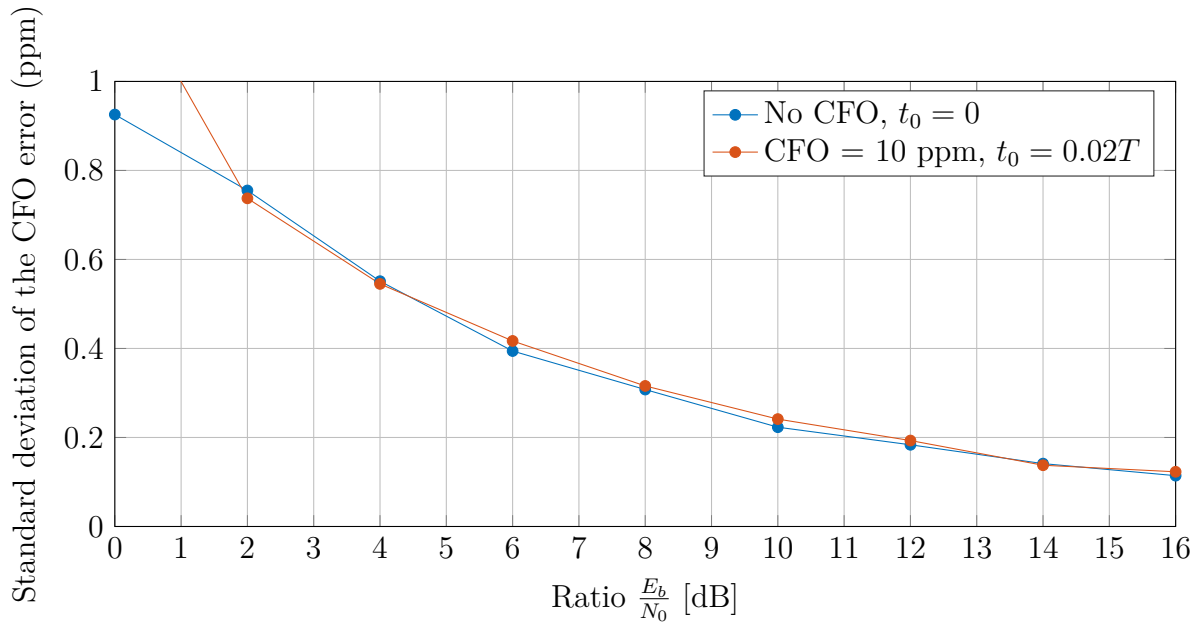Figure 17: Impact of the CFO and the sample time shift on the CFO estimation

Figure 18: Impact of the CFO and the sample time shift on the CFO estimation

4.2 Questions

4.2.1 Questions regarding the simulation

Derive analytically the baseband model of the channel including the synchronisation errors.

Assuming that the oscillators from the emitter and the receiver differ from each other by a pulsation $\Delta\omega$ and a phase shift ϕ, the signal at the receiver can be written as a function of the complex envelope of the emitted signal $e_s(t)$ (neglecting the noise):

$$r(t) = \Re\left\{e_s(t)\right\}\cos(\omega_c t) + \Im\left\{e_s(t)\right\}\sin(\omega_c t)$$

The complex envelope must be computed to go to the baseband model.

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

Using the Simpsons rules and lowpass filtering out the high frequency components, one can find

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

Hence, the synchronisation errors can be modelled by a multiplication of the baseband model by $e^{j\Delta\omega t + \phi}$.

How do you separate the impact of the carrier phase drift and ISI due to the CFO in your simulation?

The linearly increasing phase shift is manually perfectly compensated after the convolution with the second halfroot filter, taking into account the discarded samples of the convolution.

How do you simulate the sampling time shift in practice?

First, the sampling frequency is significantly increased to increase the accuracy. The sampling time shift is then implemented as a shift in the indexes when downsampling the received signal.

How do you select the simulated E_b/N_0 ratio?

The SNR should be sufficiently high so that the various algorithms in the communication chain converge. However, a noiseless simulation is not realistic. The SNR is thus fixed to typical values ranging from 5 to 10 dB.

How do you select the lengths of the pilot and data sequences?

The length of a data sequence is selected to ensure a correct phase interpolation between two pilot sequences. If it is too long, the phase interpolation is may be incorrect since $e^{j(x+k2\pi)} = e^{jx}$ and the residual CFO may be incorrectly determined. The length of the pilot should be sufficiently long to get a correct estimate of the ToA and the CFO. As shown in the results of the simulations, the remaining CFO at SNR greater than 5dB is sufficiently low when $N > 20$. However, the length and rate of repetitions of the pilot should not be too high because they reduce the channel throughput.

4.2.2 Questions regarding the communication system

In which order are the synchronisation effects estimated and compensated? Why?

The Gardner's algorithm being robust to CFO, it is first used to correct the sampling time shift. Then the CFO and the ToA are handled by the frame acquisition. Between the the pilot sequences, the residual CFO is approximated by linear interpolation.

Explain intuitively how the error is computed in the Gardner algorithm. Why is the Gardner algorithm robust to CFO?

At a given time, the estimation of the time shift is obtained from the previous estimation weighted by a certain factor. This factor depends on the middle sample between the two time steps. The direction of the correction is given by the sign of the middle value, and the magnitude of the correction is given by its magnitude. The correction will be very low if a zero crossing happens. For example, if the sampling happens too late for a downwards crossing, the value of the correction factor will be negative.

By looking at the mathematical form of the feedback implemented in the Gardner algorithm, it appears that for reasonable values of the CFO, the phases will nearly cancel each other thanks to the conjugate operator.

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 usual cross-correlator is nearly optimal according to the ML estimator but it necessitates a computationally heavy 2D exhaustive search in order to be robust to CFO. Instead, the differential cross-correlator allows for a solution of lower complexity which first estimates the ToA then the CFO.

It is not interesting to start the summation at $k = 0$ since the term $D_0[n]$ does not carry any information other than the power of the window.

Are the frame and frequency acquisition algorithms optimal? If yes, give the optimisation criterion.

The algorithm is not optimal because the optimisation is not done for both the ToA and the CFO at the same time. However, it is sufficient when both CFO and the noise have reasonable values, as it is the case in practical applications.

The optimisation criterion is the maximum likelihood of observing the received symbol y knowing the pilot and the CFO.