Project Assignment
EQ2310 Digital Communications

Analysis and Simulation of a QPSK System

(Submitted By:- Anamitra Bhar, anamitra@kth.se ; Naga Satya Sai Surya Raghava Rathan Appana, nsssrrap@kth.se)

---

1 Abstract

The performance parameters of the QPSK system in the channel are investigated in this paper, including bit error probability, synchronization, carrier phase estimation, eye diagram, spectrum characteristics, and so on. The main objective done in this study is to complete the four MATLAB functions required to realize the system simulation. This study investigates the comparison of theoretical and simulated BER performance in the Problem Areas section, as well as four other areas for discussion: signal constellation, power spectrum density, eye diagram, and the effect of training sequence length.

2 Background and Problem Formulation

This paper considers the QPSK system, which includes a bit source, transmitter, channel, receiver, and a bit sink. The bit source generates a stream of data bits that will be transmitted by a transmitter. In most simulations, random bit generators are employed as bit sources. This is also true of the article. The transmitter transforms bits to QPSK sign and optionally performs pulse shaping and up conversion. The output of the transmitter is routed through a channel, which is an AWGN channel in its most basic form. However, in some cases, multipath channels will be employed, introducing inter-symbol interference (ISI). The output of the receiver block is received by the receiver block. channel and goes via bandpass filtering, down conversion, and synchronization, sampling, and phase estimation on sampling, matched filtering, and sampling the signal received by the channel. Following the receiver's estimation of the timing and it demodulates the received QPSK symbols into information bits during the initialization phase. which are fed into the bit receiver. In most simulation environments, to collect statistics, bit receiver simply counts the number of mistakes that occur. It can be used to investigate the system's performance. [1][2]

The initial task in this project is to complete some missing MATLAB functions. These functions are as follows:

i. d = qpsk(b)
(A QPSK modulator, mapping pairs of bits 0, 1 into complex valued symbols.)
ii. tsamp = sync(mf, b train, Q, t start, t end)
(Determines when to sample the matched filter outputs.)
iii. phihat = phase estimation(r, b train)
(Phase estimator operating on the received baseband signal.)
iv. bhat = detect(r)
(Determines the received bits given a received sequence of (phase corrected) QPSK symbols. Gray coding of the individual bits is assumed.)

After the successful execution of the MATLAB code the following problem areas were taken into consideration and answered:

i. BER Performance
(Given a series of QPSK symbols that have been received, ascertain the received bits.)
ii. Signal Constellation
(Study the signal constellation in the receiver for various values of E_b/N_0. What are the implications from an error in the phase estimate?)
iii. Power Spectrum Density
(Plot the power density spectrum of the transmitted signal. How is it affected if a root raised cosine pulse is chosen instead of a rectangular pulse shape?)
iv. Eye Diagram
(Plot the eye diagram and study it for both the AWGN channel and for the simple two-path ISI channel. What implications does the ISI have on BER?)
v. Length of the Training Sequence
(Examine the performance of phase estimation and synchronization for various training sequence lengths.)

3 Methodology

A brief description on how the abovementioned five problem areas were addressed is as follows:

i. BER Performance - Plot the expression we have determined and compare it to the simulation's estimated BER.
ii. Signal Constellation - The constellation diagram will alter as E_b/N_0 is scaled up or down.
iii. Power Density Spectrum - The root raised cosine pulse and the rectangular pulse are chosen for comparison since the resulting QPSK symbols must go through an optional pulse shaping filter.
iv. Eye Diagram - The oscilloscope display is utilized to illustrate the effects of channel noise. When SNR is 10dB, contrast the eye diagrams of the two paths of the ISI channel and the AWGN channel.
v. Length of the Training Sequence - Study how the synchronization error and phase estimation error vary as a result of changing the training bit length.

4 Results

Some missing functions must be filled in before the QPSK system's properties may be investigated. In addition, certain other tasks need to be finished, like graphing the BER and the eye diagram. The appendix contains the specific code.

4.1 BER Performance

The QPSK system's BER has the equation

$$BER = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{1}$$

Phase estimation and synchronization are then utilized to send data with rectangular pulse shapes over an AWGN channel. Finally, we collect the data bits at the receiver and determine the BER of simulation by comparing the original and received data. Equation 1 is used to compute the theoretical BER and compare it to the BER obtained through simulation with SNR ranging from 0 dB to 10 dB. The outcomes are displayed in Figure 1. It is clear that the BER curve in theory and simulation closely correspond. As SNR rises, the BER will decline. This pattern suggests that higher SNR contributes to system stabilization and error reduction.

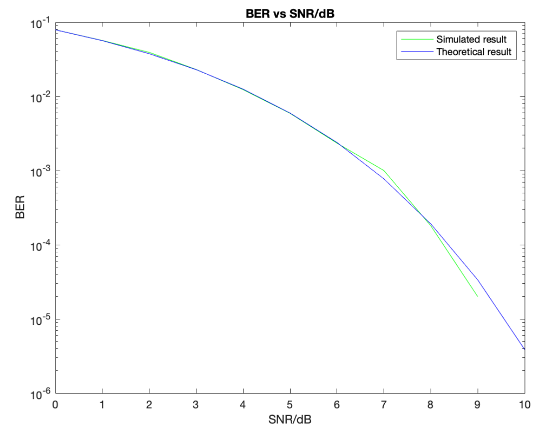
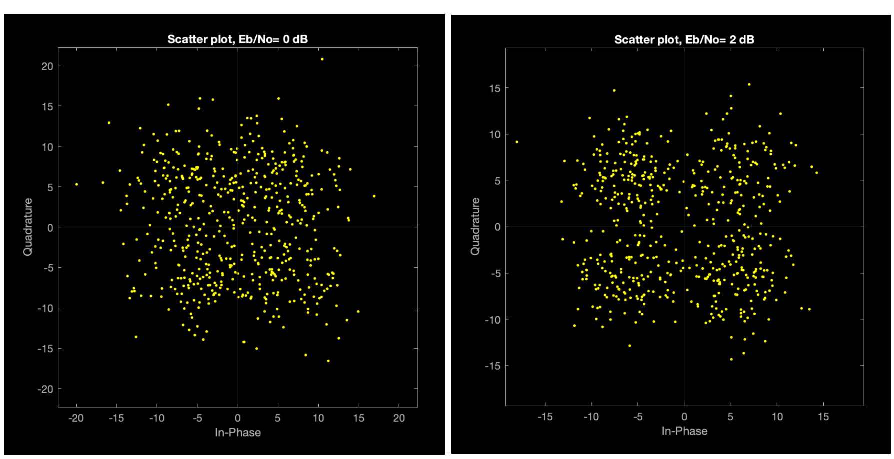


Figure 1: BER Performance Graph

4.2 Signal Constellation

Plots of the QPSK constellation are created using data at various SNRs. The outcomes are displayed in Figure 2. In order to more clearly see the effects of SNR, we chose 4 constellation maps at various SNR levels. The QPSK constellation should be centred on the designated four spots. These constellations will, however, stray from their centres because to disturbance. The constellation will be more dispersed the lower the SNR. It is clear from the plot with SNR = 0 that the noise leaves the constellations looking highly dispersed. There are numerous spots close to the edges of the identifying symbols that are quite simple to detect incorrectly.

Noise is only one of the causes of detection problems, though. Unknown phase offsets caused by the channel and various receiver filters will raise the likelihood of detection errors. However, the same outcomes can result from a phase estimation mistake. In order to successfully demodulate the signal, it is crucial to estimate the offset accurately.



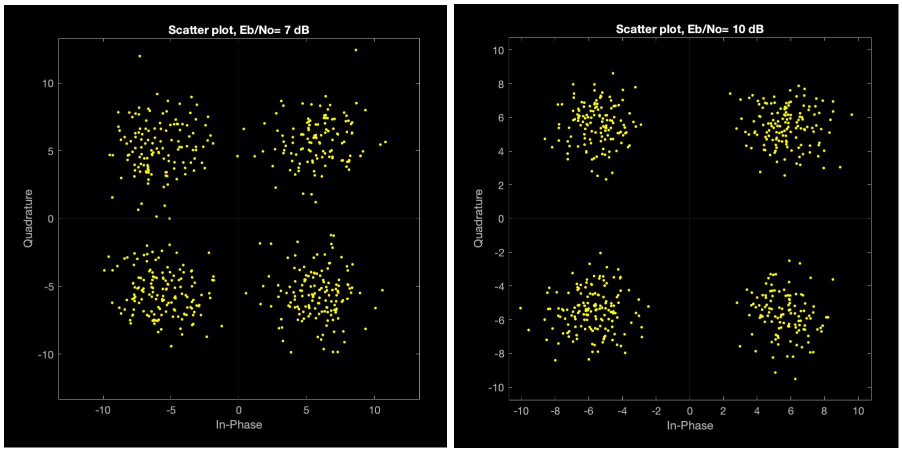


Figure 2: Signal Constellations by varying E_b/N_0

4.3 Power Spectrum Density

Figure 3 displays the power density spectrum for rectangular and root raise cosine (RRC) pulse shapes using the MATLAB function "pwelch". [3] To get rid of inter-symbol interference, employ RRC. As can be observed, the transmission bandwidth is reduced and the high frequency component that exists under rectangular pulse situations is filtered out when RRC is used.

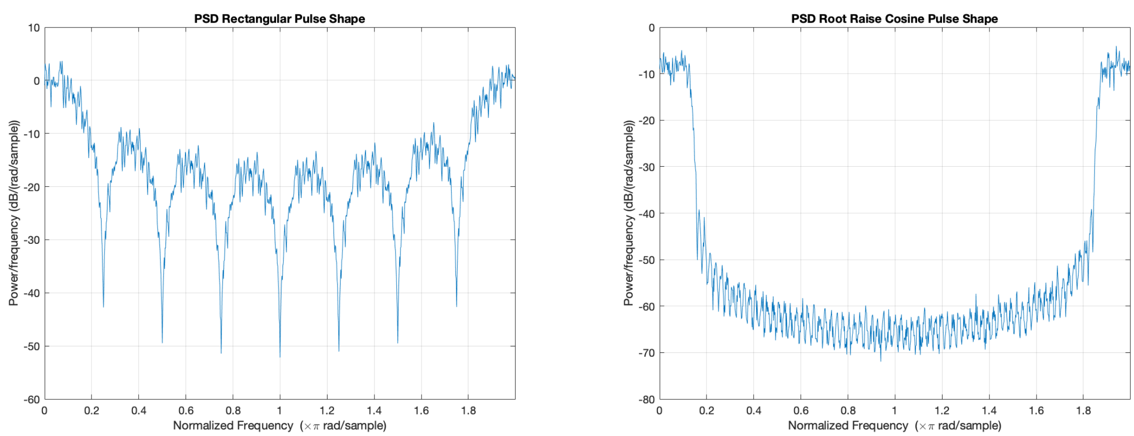
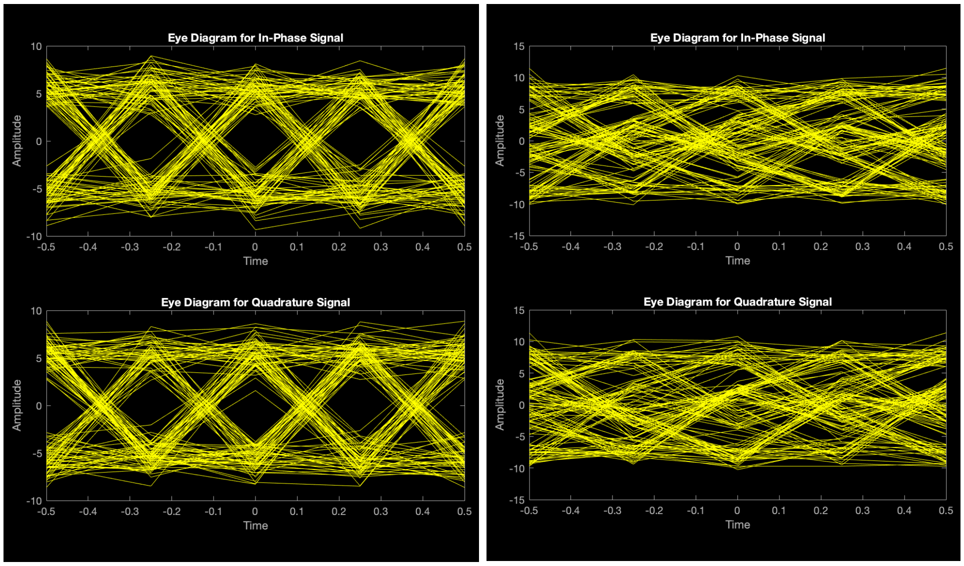


Figure 3: Power Spectral Densities

4.4 Eye Diagram

The eye diagrams for the two-path ISI channel and the AWGN channel are shown in Figure 4. The eye diagram is noticeably distorted by ISI brought on by multipath. Figures 5, 6, and 7 show how, respectively, phase estimation, BER performance, and synchronization accuracy are impacted by ISI. As opposed to the AWGN channel, the ISI channel performs worse in terms of BER, phase estimation, and synchronization accuracy, and produces poorer eye diagram results. It is evident that, especially when SNR is high, the BER of two-path ISI channels is significantly higher than that of AWGN channels. Furthermore, even if phase estimation error does exist in the AWGN channel, it is still less severe than in the two-path ISI channel. Furthermore, both of their synchronization faults are negligibly small when the size of the training sequence is set to 100. However, it becomes obvious that accuracy in the two-path ISI channel is lower when the size is reduced to 10.



(a) AWGN Channel (b) Two-path ISI Channel

Figure 4: Eye Diagram

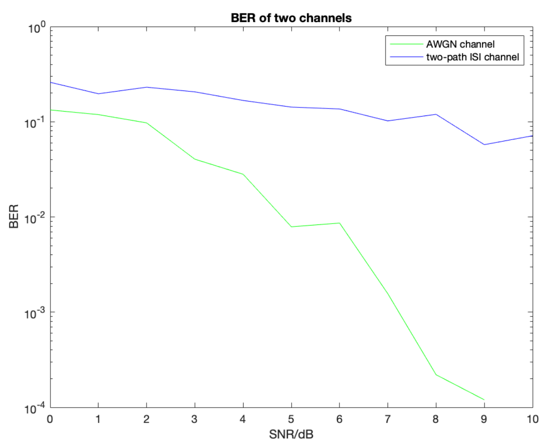


Figure 5: BER of the two channels

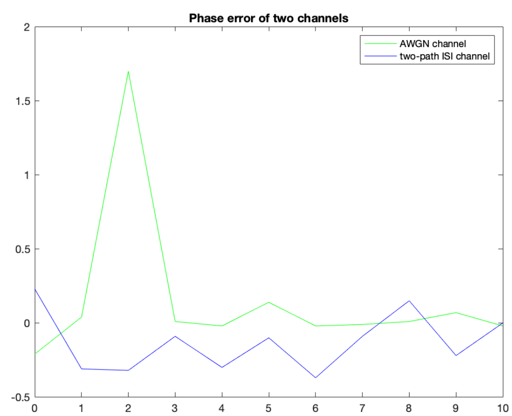


Figure 6: Phase Error of the two channels

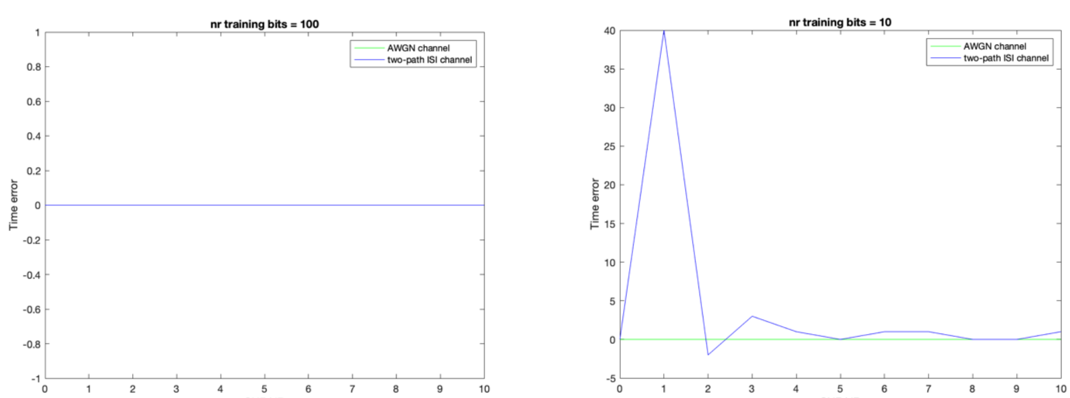


Figure 7: Synchronization Error of the two channels

4.5 Length of the Training Sequence

To calculate the estimation error for each duration, we fix the E_b/N_0 as 5dB and set the training sequence length to range from 2 to 100. The outcome shown in Figure 8 demonstrates how training sequence duration affects phase estimation and synchronization performance.

For instance, the plot for synchronization error demonstrates that when the training length is less than 16, the synchronization error is rather high, however as the training length increases, it approaches zero. Another example is provided by the plot of phase error, which demonstrates how the phase estimate error decreases as the training length increases and is quite significant when it is less than 18. Then, when we varied E_b/N_0 between 0 and 20dB, the outcome essentially confirmed our earlier findings. Therefore, we may say that a training sequence length of 20 can accomplish quite effective phase estimation and synchronization, while the subsequent sequence may be useless.

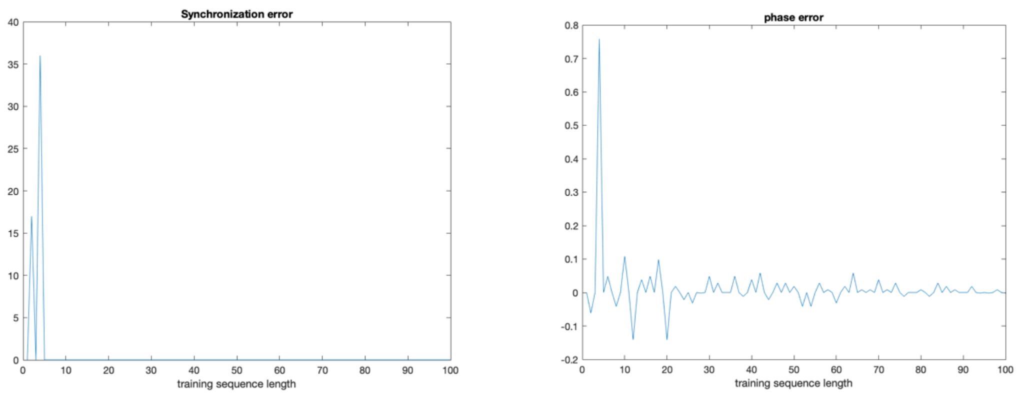


Figure 8: Synchronization and Phase Error under varying Training Sequence Length

5 Conclusions

We succeeded in simulating a QPSK system in this research. With flawless phase estimation and synchronization, we first looked at how the BER performed as a function of E_b/N_0 and discovered that the simulation results completely corroborated the theoretical findings. Then, under various conditions, we plotted the signal constellation, power density spectrum, and eye diagram. Additionally, we statistically examined how different channel types and the length of the training cycle affected phase estimation and synchronization. In sum, we improved our understanding of communication theory for future study.

Appendix

Our implementation of the missing MATLAB functions are as follows:

```
function d = qpsk(b)
    N = length(b);
    d = zeros(1, N/2);
    b = reshape(b, 2, N/2);
    d = (1 - 2 * b(1,:)) + 1j * (1 - 2 * b(2,:));
    d = d / sqrt(2);
end


function t_samp = sync(mf, b_train, Q, t_start, t_end)
time=t_end-t_start;
r=zeros(1,time+1);
qpsk_train=qpsk(b_train);
len=length(qpsk_train);
for i=t_start:t_end
a=mf(i+((0:len-1).*Q));
r(i-t_start+1)=abs(a*qpsk_train');
end
[M,I]=max(r);
t_samp=I+t_start-1;
end


function phihat = phase_estimation(r, b_train)
    qpsk_train = qpsk(b_train);
    len = length(qpsk_train);
    r_train = r(1:len);
    phihat = angle(sum(r_train .* conj(qpsk_train)));
end


function bhat = detect(r)
    N = length(r);
    bhat = zeros(1, 2 * N);
    for i = 1:N
        if real(r(i)) >= 0
            bhat(2 * i - 1) = 0;
        else
            bhat(2 * i - 1) = 1;
        end
        if imag(r(i)) >= 0
            bhat(2 * i) = 0;
        else
            bhat(2 * i) = 1;
        end
    end
end
```

References

[1] Madhow, U. (2008). Fundamentals of Digital Communication. Cambridge University Press.

[2] Proakis, J. G., & Salehi, M. (2005). Digital Communications (5th ed.). New York, NY: McGraw-Hill.

[3] MathWorks. (n.d.). pwelch. [Online]. Available: https://www.mathworks.com/help/signal/ref/pwelch.html [Accessed: 08-Feb-2023].