Digital Communication Lab

Laboratory report submitted for the partial fulfillment
of the requirements for the degree of

Bachelor of Technology
in
Electronics and Communication Engineering

by

Mohit Akhouri - 19ucc023

Course Coordinator
Dr. Nikhil Sharma

Department of Electronics and Communication Engineering
The LNM Institute of Information Technology, Jaipur

September 2021

Contents

Chapter *6*

Experiment - 6

6.1 Name of the Experiment

Performance analysis of M-ary phase-shift keying modulation scheme over AWGN channel

6.2 Software Used

- MATLAB
- Simulink

6.3 Theory

6.3.1 About AWGN channel :

Additive white Gaussian noise (AWGN) is a basic noise model used in information theory to mimic the effect of many random processes that occur in nature. The modifiers denote specific characteristics:

- Additive : it is added to any noise that might be intrinsic to the information system.
- White : refers to the idea that it has **uniform power** across the frequency band for the information system. It is an analogy to the color white which has **uniform emissions** at all frequencies in the **visible spectrum**.
- Gaussian : because it has a normal distribution in the time domain with an average time domain value of zero and variance σ^2.

Wideband noise comes from many natural noise sources, such as the thermal vibrations of atoms in conductors (referred to as thermal noise or Johnson–Nyquist noise), shot noise, black-body radiation from the earth and other warm objects, and from celestial sources such as the Sun. The **central limit theorem** of probability theory indicates that the summation of many random processes will tend to have distribution called **Gaussian** or **Normal**.

6.3.2 About Bit error rate of M-ary Phase Shift Keying :

M-ary phase-shift keying (MPSK) is employed in some of the digital cellular standards and communication geostationary satellite systems. MPSK employs a set of M equal-energy signals to represent M **equiprobable symbols**. This constant energy restriction (i.e., the constant envelope constraint) warrants a circular constellation for the signal points. In MPSK, the phase of the carrier takes on one of M possible values $\frac{2\pi(i-1)}{M}$, where i = 1, 2, ...,M. The MPSK signal set is thus analytically given by:

$$S_it = \sqrt{\frac{2E_i}{T_s}} cos\left(2\pi f_c t - \frac{2\pi i - 1}{M}\right) \tag{6.1}$$

In the above equation , **t** ranges from 0 to T_s and values of **i** are from 1 to M. We assume a **uniform spacing** of phase values, i.e., the phase separation between any two adjacent signal points is **constant**. The Energy associated with the MPSK signal set is as follows :

$$E_i = E_s \tag{6.2}$$

In the above equation , **i** ranges from 1 to M. E_s represents the **average energy per symbol**.

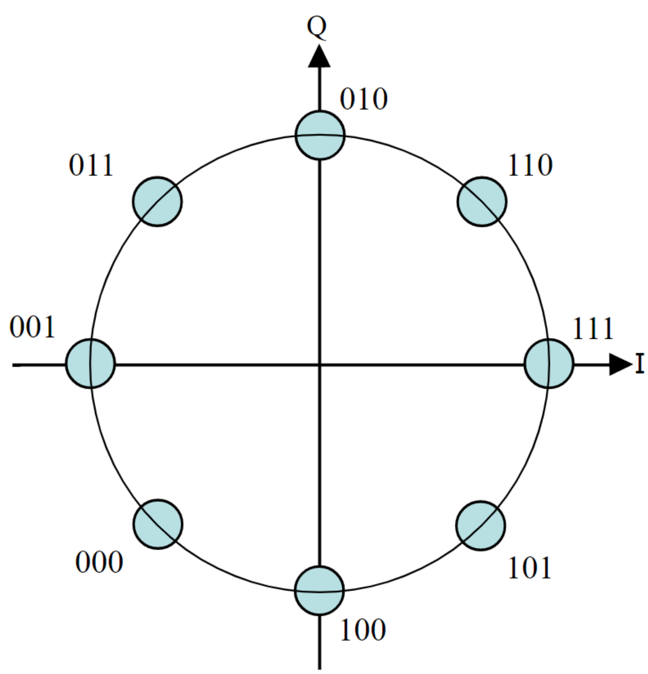

Figure 6.1 Constellation diagram of 8-ary Phase Shift Keying

The **MPSK signal set** is characterized by a **two dimensional signal space** and **M message points** :

$$S_it = \sqrt{E_i} cos\left(\frac{2\pi(i-1)}{M}\right)\phi_1 t + \sqrt{E_i} sin\left(\frac{2\pi(i-1)}{M}\right)\phi_2 t \tag{6.3}$$

The **average symbol error probability** be tightly approximated as follows :

$$P_{SER-MPSK} \approx 2Q\sqrt{\frac{2E_s}{N_o}} sin\left(\frac{\pi}{M}\right) \tag{6.4}$$

6.3.3 About Bit error rate of 4-QPSK Modulation :

Quadrature Phase Shift Keying (QPSK) is a form of PSK which uses a combination of two bits (00, 01, 10 & 11). Each of this bit combination is represented by **four possible carrier phase shifts** (0, 90, 180 & 270). With QPSK **twice** as much information as ordinary PSK can be transmitted using the same bandwidth. In QPSK the **amplitude of carrier** remains **constant** for all symbols. QPSK modulation consists of **two BPSK modulation** on in-phase and **quadrature components** of the signal.

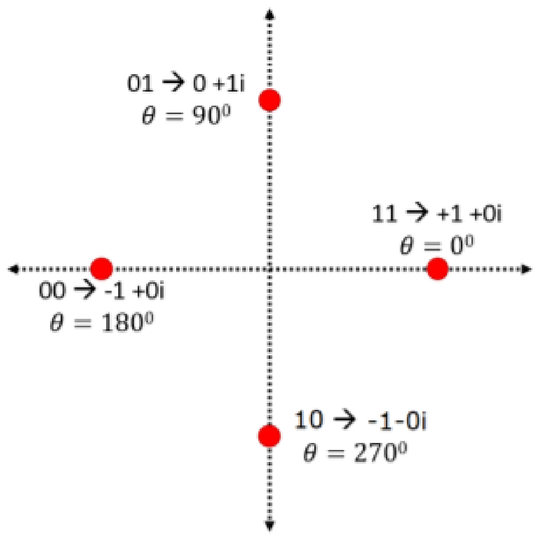

Figure 6.2 Constellation diagram of 4-QPSK Modulation

6.3.3.1 Symbol Error rate of 4-QPSK Modulation :

The BER of each branch is the same as BPSK :

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_o}}\right) \tag{6.5}$$

The **symbol probability of error (SER)** is the probability of either branch has a bit error:

$$P_s = 1 - \left[1 - Q\left(\sqrt{\frac{2E_b}{N_o}}\right)\right]^2 \tag{6.6}$$

Since the **symbol energy** is **split** between the **two in-phase and quadrature components**, $\frac{E_s}{N_o} = \frac{2E_b}{N_o}$ and we have:

$$P_s = 1 - \left[1 - Q\left(\sqrt{\frac{E_s}{N_o}}\right)\right]^2 \tag{6.7}$$

6.4 Code and Results

6.4.1 BER and SER of M-ary Phase Shift Keying and 4-QPSK Modulation :

```
% 19ucc023
% Mohit Akhouri
% Observation 1 - Modulation order and Probability of Error for M-ary
 PSK

clc;
clear all;
close all;

size1 = 10; % intializing the size for BER and SER
size2 = 10000; % intializing the size for signal x[n]

SER_Practical = zeros(1,size1); % Initializing Practical SER array
SER_Theoretical = zeros(1,size1); % Initializing Theoretical SER array

BER_Real_part = zeros(1,size1); % Initializing Real part of
 BER_Practical
BER_Imag_part = zeros(1,size1); % Initializing Imaginary part of
 BER_Practical

BER_Practical = zeros(1,size1); % Initializing Practical BER array
BER_Theoretical = zeros(1,size1); % Initializing Theoretical BER array

x_real = zeros(1,size2); % Initializing Real part of x[n]
x_img = zeros(1,size2); % Initializing Imaginary part of x[n]
x = zeros(1,size2); % Intializing input signal x[n]
y = zeros(1,size2); % Initializing Output signal y[n]

% Main loop algorithm for calculation of transmitted signal x[n]
for i=1:size2

    rnd1 = rand(); % random value 1 generation
    rnd2 = rand(); % random value 2 generation

    % Constructing the real part of signal on the basis of decision
    if(rnd1 > 0.5)
        x_real(i)=1;
    else
        x_real(i)=-1;
    end

    % Constructing the Imaginary part of signal on the basis of
 decision
    if(rnd2 > 0.5)
        x_img(i)=1;
    else
        x_img(i)=-1;
    end
end

x = x_real + (1j * x_img); % Overall transmitted signal x[n]
```

Figure 6.3 Part 1 of the Code for Observation 1

```
SNR_dB = 0:9; % defining the range of Signal to Noise Ratio ( Measured
 in dB )

% Main loop algorithm for calculation of x[n],y[n], noise "n"
% and calculation of theoretical and practical BER and SER
for i=1:length(SNR_dB)

    SNR=10^((i-1)/10);
    N = 1/SNR;
    M = sqrt(N/2);

    n=zeros(1,size2); % Initializing noise signal n

    % loop for calculation of noise signal
    for j=1:size2
        n(j) = M*randn() + ( 1j * M * randn());
    end

    % loop for calculation of received AWGN + x[n] signal
    for j=1:size2
        y(j) = x(j) + n(j);
    end

    yn = zeros(1,size2);
    y_real = zeros(1,size2);
    y_img = zeros(1,size2);

    % Main Loop algorithm for ML-Detection of M-ary and QPSK
 Modulation
    for j=1:size2
        if(real(y(j)) >= 0)
            y_real(j) = 1;
        else
            y_real(j) = -1;
        end

        if(imag(y(j))>=0)
            y_img(j) = 1;
        else
            y_img(j) = -1;
        end

        yn = y_real + (1j * y_img);
    end

    % Comparing the transmitted and received message signal
    % and calculating the Practical BER and SER
    for j=1:size2
        if(x(j)~=yn(j))
            SER_Practical(i) = SER_Practical(i) + 1;
        end

        if(real(x(j)) ~= real(yn(j)))
            BER_Real_part(i) = BER_Real_part(i) + 1;
```

Figure 6.4 Part 2 of the Code for Observation 1

```
        end

        if(imag(x(j)) ~= imag(yn(j)))
            BER_Imag_part(i) = BER_Imag_part(i) + 1;
        end
    end

    BER_Practical(i) = ((BER_Real_part(i)/size2) + (BER_Imag_part(i)/
size2))/2;
    BER_Theoretical(i) = qfunc(sqrt(2/N));

    SER_Practical(i) = SER_Practical(i)/size2;
    SER_Theoretical(i) = 2 * qfunc(sqrt(2/N));
end

SER_matrix = zeros(10,5); % Matrix for storing SER values for
 different modulation orders
M = [2,4,8,16,32]; % array of modulation orders M

% Loop for calculation of SER for different modulation order M
for i=1:length(SNR_dB)
    SNR = 10^((i-1)/10);
    N = 1/SNR;

    for m = 1:length(M)
        SER_matrix(i,m) = 2 * qfunc(sqrt(2/N) * sin(pi/M(m)));
    end

end

% Displaying the SER matrix
disp('SER vs. Modulation order matrix is given as:');
disp(SER_matrix);

% Plot of SER for different modulation order
figure;
semilogy(SNR_dB,SER_matrix(:,1),'color','blue');
hold on;
semilogy(SNR_dB,SER_matrix(:,2),'color','black');
semilogy(SNR_dB,SER_matrix(:,3),'color','red');
semilogy(SNR_dB,SER_matrix(:,4),'color','magenta');
semilogy(SNR_dB,SER_matrix(:,5),'color','cyan');
ylabel('SER ->');
xlabel('SNR(dB) ->');
title('19ucc023 - Mohit Akhouri','Plots of SER for different values of
 Modulation order (M) for M-ary Phase Shift Keying');
legend('SER for M = 2','SER for M = 4','SER for M = 8','SER for M =
 16','SER for M = 32');
grid on;
hold off;

% Plots of practical and theoretical SER vs. SNR ( in dB )
figure;
semilogy(SNR_dB,SER_Practical,'Color','blue');
```

Figure 6.5 Part 3 of the Code for Observation 1

```
hold on;
semilogy(SNR_dB,SER_Theoretical,'Color','red');
xlabel('SNR(dB) ->');
ylabel("SER ->");
title('19ucc023 - Mohit Akhouri','Plots of Practical and Theoretical
 SER vs. SNR (dB) for 4-QPSK Modulation');
legend('Practical SER','Theoretical SER');
grid on;
hold off;

% Plots of practical and theoretical BER vs. SNR ( in dB )
figure;
semilogy(SNR_dB,BER_Practical,'Color','blue');
hold on;
semilogy(SNR_dB,BER_Theoretical,'Color','red');
xlabel('SNR(dB) ->');
ylabel("BER ->");
title('19ucc023 - Mohit Akhouri','Plots of Practical and Theoretical
 BER vs. BER (dB) for 4-QPSK Modulation');
legend('Practical BER','Theoretical SER');
grid on;
hold off;
```

Published with MATLAB® R2020b

Figure 6.6 Part 4 of the Code for Observation 1

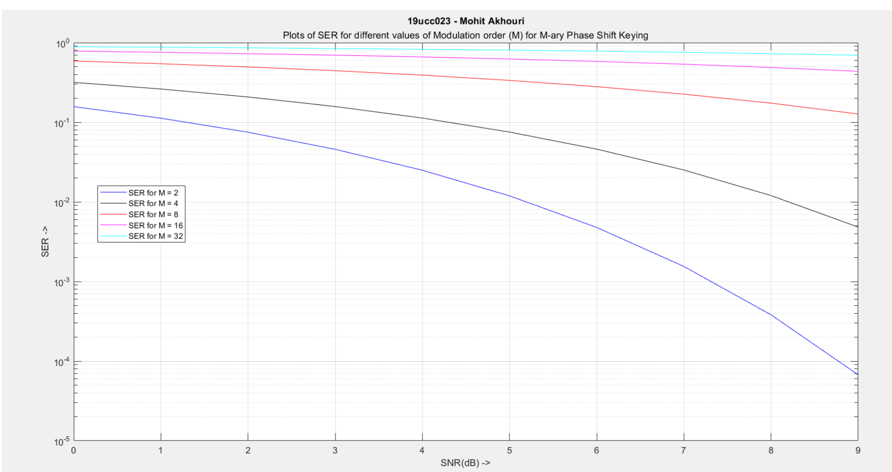

Figure 6.7 Plots of SER for different values of Modulation order M

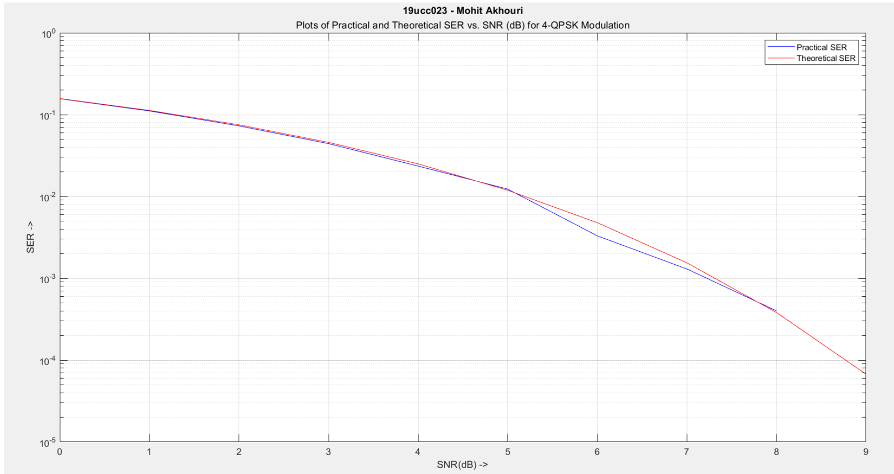

Figure 6.8 Plots of Theoretical and Practical SER vs. SNR (dB)

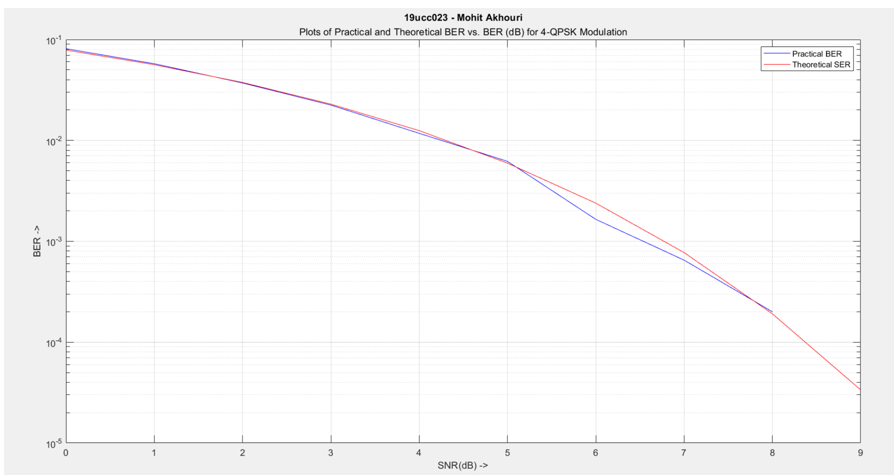

Figure 6.9 Plots of Theoretical and Practical BER vs. SNR (dB)

6.4.2 Relationship between Probability of error and Modulation order :

```
% 19ucc023
% Mohit Akhouri
% Observation 2 - Calculation and Plotting of Probability of Error vs.
% Modulation order

% This code will first calculate the Probability of error for
 different
% modulation order and Plot the graphs between them

clc;
clear all;
close all;

val1 = 5; % SNR = 5 dB stored in val1
val2 = 10; % SNR = 10 dB stored in val2
M=[2 4 8 16 32]; % Initializing the modulation order (M) array

PBE_for_SNR_5 = zeros(1,5); % Initializing array 1 for PBE for SNR = 5
 dB
PBE_for_SNR_10 = zeros(1,5); % Initializing array 2 for PBE for SNR =
 10 dB

% Calculation of Probability of Error for SNR = 5 dB

SNR_1 = 10.^(val1/10); % Calculating SNR 1

for i=1:5
    PBE_for_SNR_5(i) = 2 * qfunc(sqrt(SNR_1))*sin(pi/M(i));
end

% Calculation of Probability of Error for SNR = 10 dB

SNR_2 = 10.^(val2/10); % Calculating SNR 2

for i=1:5
    PBE_for_SNR_10(i) = 2*qfunc(sqrt(SNR_2))*sin(pi/M(i));
end

% Displaying the values of probability of error for different values
 of M
display('Probability of error (for SNR = 5dB) values for
 M=2,4,8,16,32 :');
display(PBE_for_SNR_5);

display('Probability of error (for SNR = 10dB) values for
 M=2,4,8,16,32 :');
display(PBE_for_SNR_10);

% Plots of Probability of error vs. Modulation order M for SNR = 5 dB
figure;
plot(PBE_for_SNR_5);
xlabel('Modulation order (M) ->');
```

Figure 6.10 Part 1 of the Code for Observation 2

```
ylabel('Probability of error ->');
title('19ucc023 - Mohit Akhouri','Probability of error vs. Modulation
 order (M) for SNR = 5 dB');
grid on;

% Plots of Probability of error vs. Modulation order M for SNR = 10 dB
figure;
plot(PBE_for_SNR_10);
xlabel('Modulation order (M) ->');
ylabel('Probability of error ->');
title('19ucc023 - Mohit Akhouri','Probability of error vs. Modulation
 order (M) for SNR = 10 dB');
grid on;
```

Published with MATLAB® R2020b

Figure 6.11 Part 2 of the Code for Observation 2

```
Probability of error (for SNR = 5dB) values for M=2,4,8,16,32 :

PBE_for_SNR_5 =

    0.0754    0.0533    0.0288    0.0147    0.0074

Probability of error (for SNR = 10dB) values for M=2,4,8,16,32 :

PBE_for_SNR_10 =

    0.0016    0.0011    0.0006    0.0003    0.0002
```

Figure 6.12 Display of values of Probability of error for different values of M

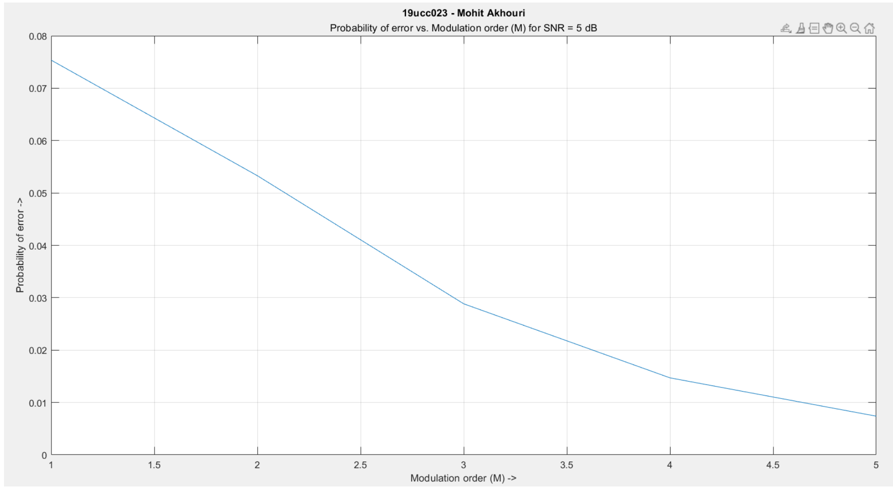

Figure 6.13 Plot of Probability of error vs. Modulation order for SNR = 5dB

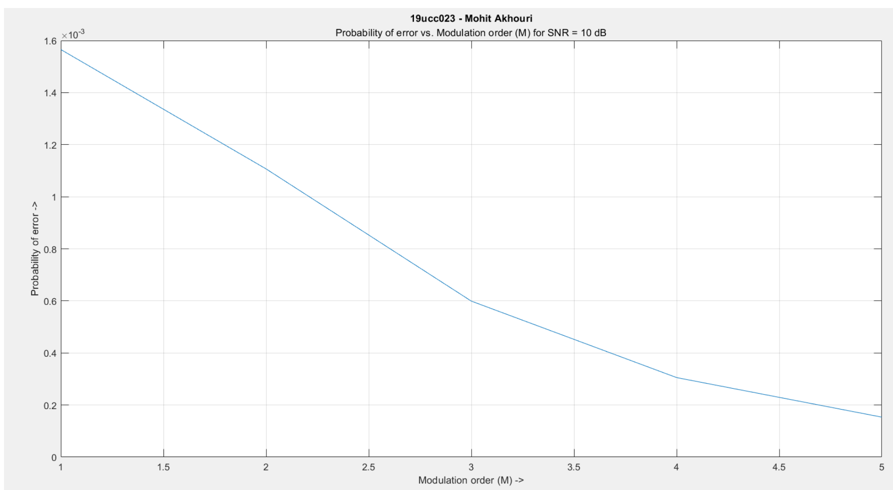

Figure 6.14 Plot of Probability of error vs. Modulation order for SNR = 10dB

6.5 Conclusion

In this experiment , we learnt about two types of Modulation **M-ary PSK Modulation** and **4-QPSK Modulation**. We analysed how to generate the M-ary PSK and 4-QPSK waveforms and calculate their Bit-Error rate and symbol error rate. We learnt about important concepts like the **AWGN noise** and how it affects the Bit error rate. We also learnt about **Q-function** and how to calculate Theoretical BER of both M-ary PSK and 4-QPSK Modulation. We implemented the codes in MATLAB and analysed the results. We also plotted the graph between BER and SNR (in dB) and also between SER and SNR (in dB) and verified the results.