King Fahd University of Petroleum & Minerals
College of Engineering Sciences
Electrical Engineering Department

EE 419 Wireless Communications

Term 212
Section 01

Term Project

A Study of the Performance of Wireless Communication over Fading Channels with Diversity Impact.

Prepared by:

Serial No.	Name								
9	Ahmed AlGhamdi								
	2	0	1	7	5	7	4	3	0
12	Abdullah AlKathiry								
	2	0	1	7	6	5	2	9	0
23	Hussain Balhareth								
	2	0	1	7	8	2	0	5	0

For:

Dr. Ali Muqaibel

9 May 2022

Abstract:

This report shows a performance simulation of a wireless communication channel. The simulation was done on MATLAB, and the programming code is explained in detail. The results are shown for various modulations types with m-ary. We discuss the results of the performance over AWGN channel and fading channel, and we investigate the effects of diversity. We then test sending an audio recording through the system.

Table of Contents

I. Introduction

Wireless communication is the most valuable communication technology that evolved in the past century. With such a technology, millions of computers and cellphones are connected without any physical links but through radiation paths, allowing satisfactory and reliable communication. However, wireless communication techniques face lots of challenges and not noise as the case with wired communication. There are three main limitations of the wireless systems [1]:

- **Path loss:**
  The attenuation in the power of the signal wave as it propagates through space due to the distance traveled. (Deterministic modeling)
- **Shadowing:**
  The fluctuation in the power of the signal wave due to obstruction with different bodies while propagating from transmitter to receiver. (Random modeling)
- **Multipath fading:**
  The variation of the attenuation in the power of the signal wave due to the receiver's mobility or other-waves reflections of multiple paths. (Random modeling)

The impact of these limitations is demonstrated in Fig.1 below.

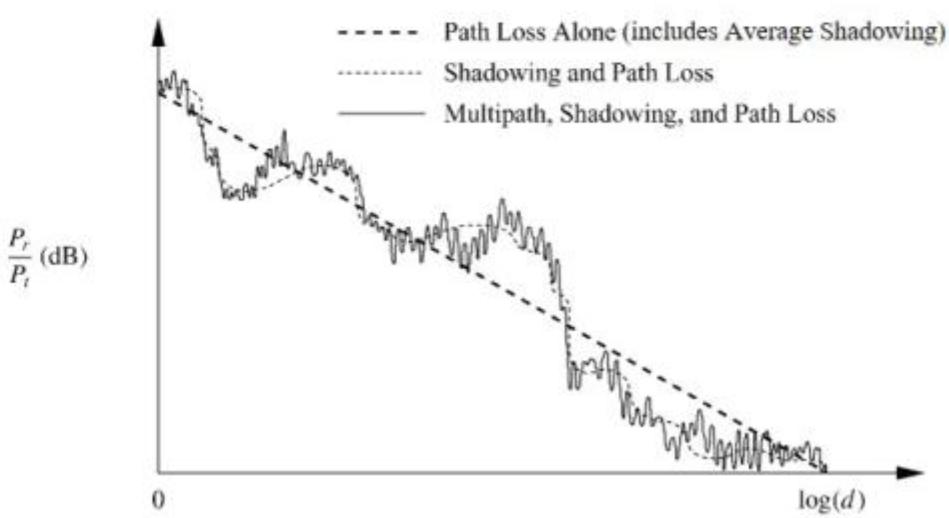

Figure 1: Received Power vs Distance with path loss, shadowing, and multipath fading effects. [2]

To reduce those limitations, several engineering and designing techniques have been used. One example is diversity, which can mitigate the effects of fading by combining the independent paths that will not experience deep fades simultaneously. The concept behind combining is transmitting data over independent paths combined in such a way to reduce the resultant signal. Frequency and time diversities are examples of diversity methods. The Frequency diversity states that versions of a signal are transmitted at different carrier frequencies and time diversity time instants [2].

Therefore, this project aims to evaluate the performance of wireless communications over fading channels and see the impact of diversity. It will start by simulating the performance of the communication system over the AWGN channel on MATLAB using the provided code with modifications shown in Appendix A. The modified code extends the simulation to M-ary communication and uses different modulation schemes. Next, the performance over fading channels will be evaluated before and after applying diversity to see the diversity impact on the performance. Lastly, the simulation will include voice communication to see the system's quality and how it is affected while trying different diversity situations.

II. Background

The communication system modulation schemes we are using are: MPAM, MQAM, and MPSK with different values of M. Each scheme has its characteristics, noise, and error modeling. The performance among the schemes is evaluated based on the Symbol Error Rate (SER) and the Bit Error Rate (BER). BER and SER provide a clear picture of the performance of the communication system.

For MPSK and MPAM modulations over AWGN channel, we have:

$$SER = erfc(\frac{d}{2}\sqrt{\left(\frac{E_b}{N_0}\right)})$$

$$BER = \frac{SER}{\log_2 M}$$

Where d is the distance between adjacent symbols, E_b is the bit energy, and N_0 is the AWGN noise.

For M-QAM modulation over AWGN channel, we have:

$$SER = 2(1 - \frac{1}{\sqrt{M}})erfc(\sqrt{EbN0});$$

For fading channels, we have:

$$BER = \frac{1}{2}(1 - \sqrt{\frac{\frac{E_b}{N_0}}{1 + \frac{E_b}{N_0}}})$$

Which applies on all modulations.

III. MATLAB Code

The MATLAB code is written in a single m file which is being divided into many sections. We will explain what is happening in every section of the code. The functions “dec2gc” and “gc2dec” are required and can be downloaded as a MATLAB add-on.

A. Simulation Settings

The MATLAB code is general; i.e. there is a single m file that simulates the performance for all combinations of parameters. In this section, we select the parameters of the simulation. We select the modulation to be MPSK, QAM, or MPAM, and we also select the number M. We select if the channel is fading or not, and we select the diversity order ant its type, MRC or EGC. Lastly, we select if we are sending an audio recording or not, and we select SNR value of the audio.

B. Producing the Bits Sequence

In this section, we generate a random stream of bits which will be the data to be sent. In case we are sending an audio recording, MATLAB will start recording audio at “fs” sampling frequency. The audio will be quantized using an 11-bit quantizer and the bits will be reshaped into a stream of bits.

C. Coordinates of the Modulation

In this part, we set up the positions of the symbols on the coordinate based on the modulation scheme. For example, 4-PSK will be assigned the values (1,0), (0,1), (-1,0), (0,-1) on the xy-coordinate. M-PAM will be assigned on the x-axis only. The coordinates will be assigned as a matrix with M rows and 2 columns. The 2 columns represent the x-axis and y-axis. For QAM, we use the function "qammod", which produce the points as complex numbers. The real part is the x-axis and the imaginary part is the y-axis.

D. Bits to Symbols

In this part of the code, the bits are assigned to the symbols of the modulation scheme. In case there are extra bits in the end that cannot form a full symbol, they will be removed. We now assign every M bits to a symbol using a for loop. The symbols are stored in the variable "y". To make the symbols gray codded, we use the function "gc2dec" to get a decimal number. For example, 00 = 1, 01 = 2, 11 = 3, 10 = 4. After that, we assign a new coordinate "coordinate_s" to be the gray codded coordinate using a loop.

E. AWGN, Fading, and Diversity

This part of the code is compute-heavy and will take some time for processing, especially if the number of bits in the bit stream is large; i.e. 10^5 bits. The fading is added in this part, then the AWGN added and diversity calculations are processed. We have applied the fading to the constellation's coordinates, not to the time domain. The diversity will be calculated from order 1 (no diversity) to the order specified in the Simulation Settings with step of 1. Then, the received symbols will be reshaped to a stream of bits. Thus, we calculate SER and BER by comparing the received symbols and bits with the transmitted ones. The section code will run in a loop to calculate the BER and SER with SNR ranging from 0 dB to 12 dB.

F. Plotting

Here, we generate the plots for the BER and SER, both simulation and theory. We use an if statement to determine the cases such fading or no fading, modulation scheme, etc... The title and legend are also general for all types of modulations. The simulation plots are shown for the known cases such as all modulations over AWGN channel, and BPSK over fading channel.

G. Play the Recording

This part reshapes the audio signal that was transmitted, decodes it, and plays the audio recording. You can this section alone to hear the audio again.

IV. Results and Discussion

For this project, there were certain tasks that were assigned to certain students. Student A is Ahmed, Student B is Abdullah, and Student C is both Hussain and Abdullah. We will now show the results in terms of probability of error for various modulations with m-ary. We begin with AWGN channel, then we simulate with fading channel, and lastly, we apply diversity to the fading channel to enhance the results.

A. AWGN Channel

MPSK modulation with M = 2, 4, 8.

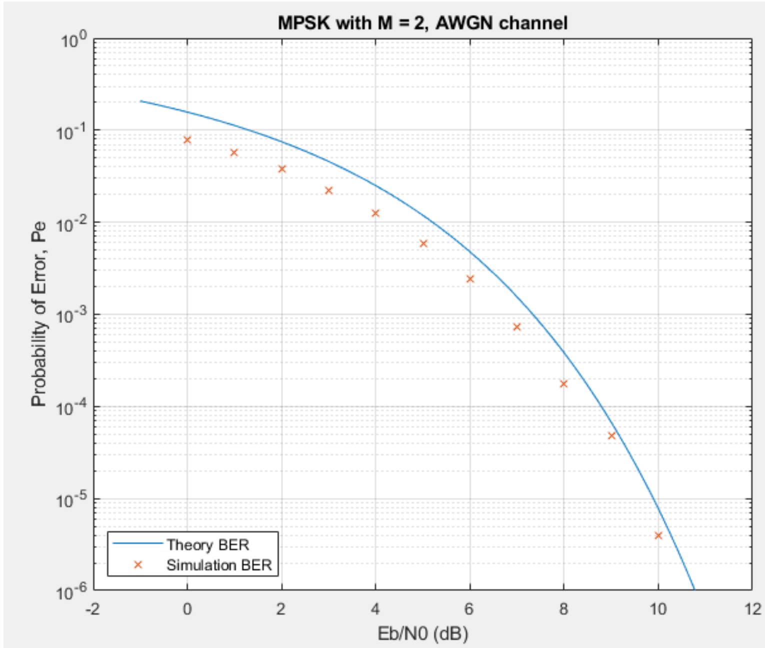

Figure 2

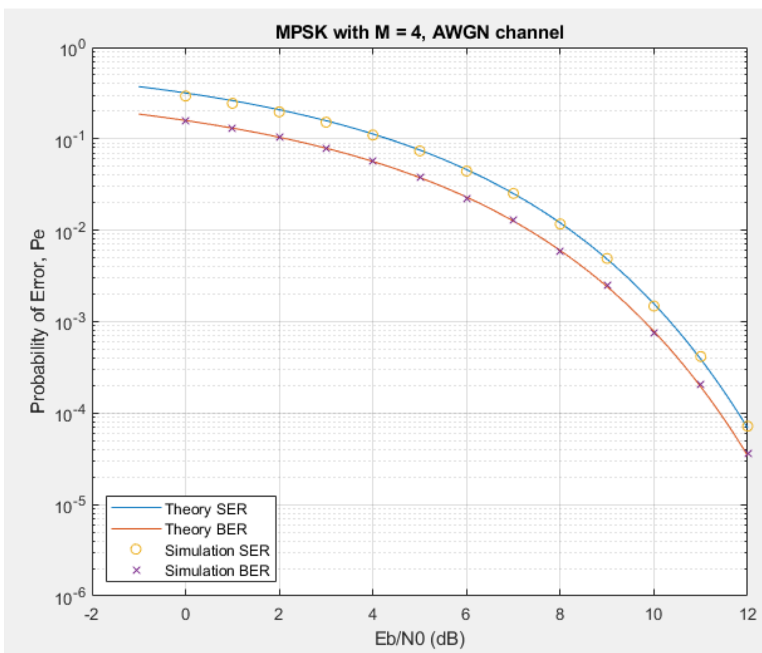

Figure 3

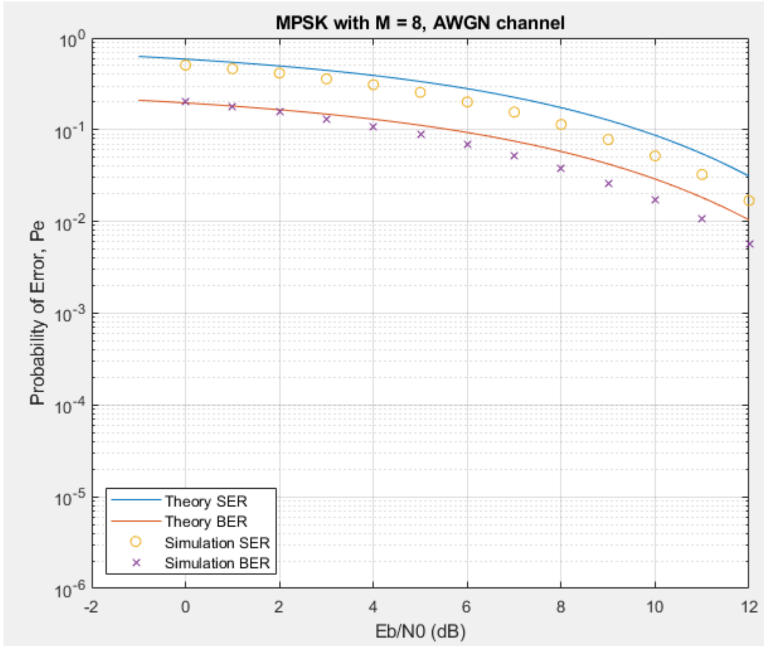

Figure 4

We can see that the simulation results meet the theoretical expectations. We can also notice that increasing M increases the error probability. This is expected since the distance between symbols is decreasing, but the power is constant.

QAM modulation with M = 8, 16.

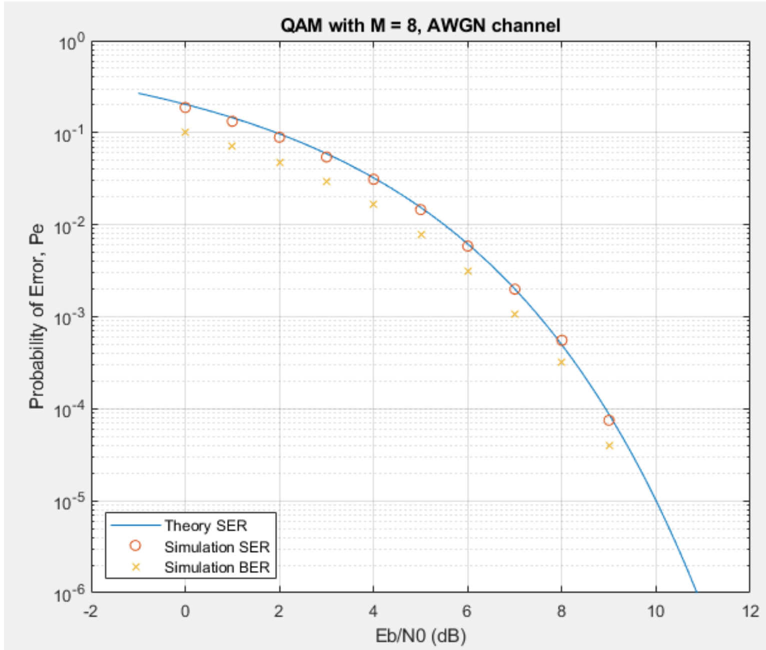

Figure 5

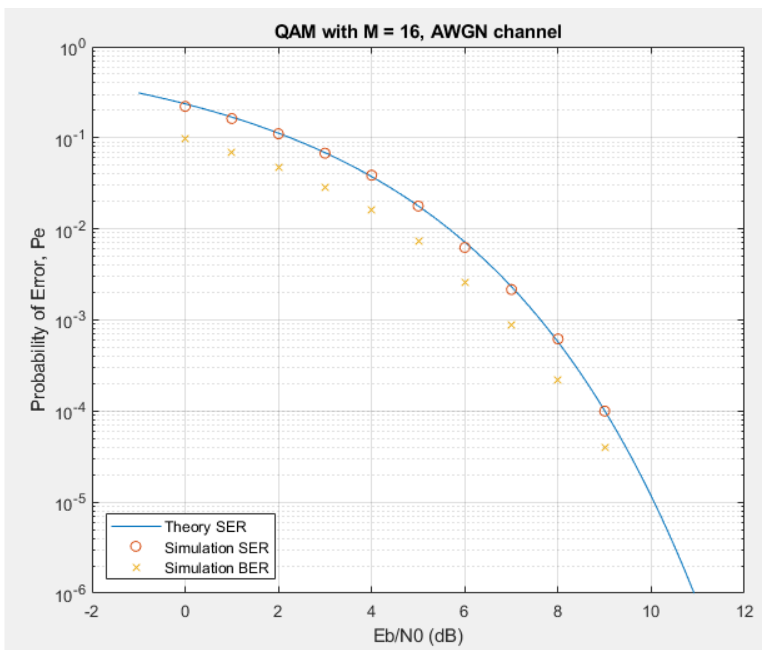

Figure 6

We can see here that the error probability does not change when increasing M. This is because the minimum distance between the symbols is constant. However, increasing M results in increasing the power as well.

MPAM modulation with M = 2, 4, 8.

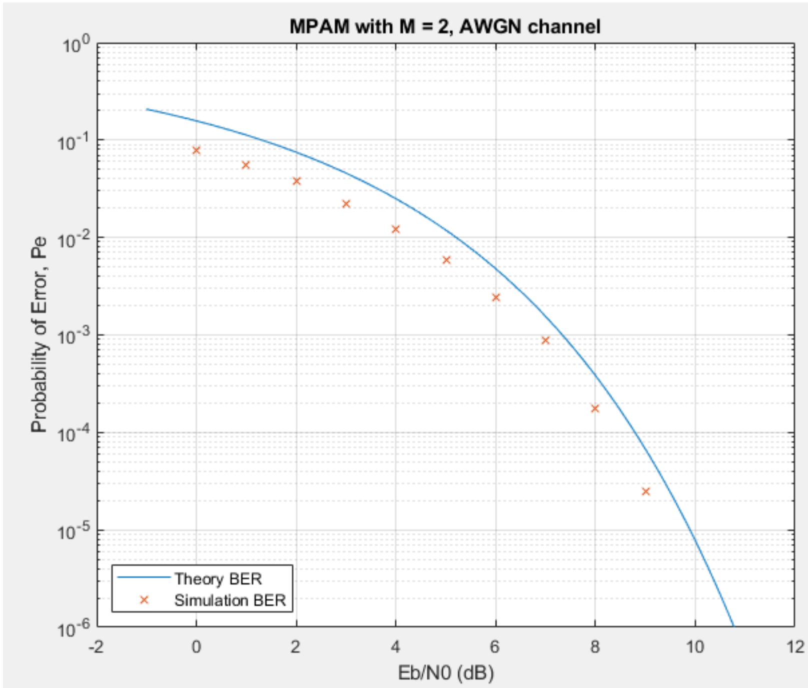

Figure 7

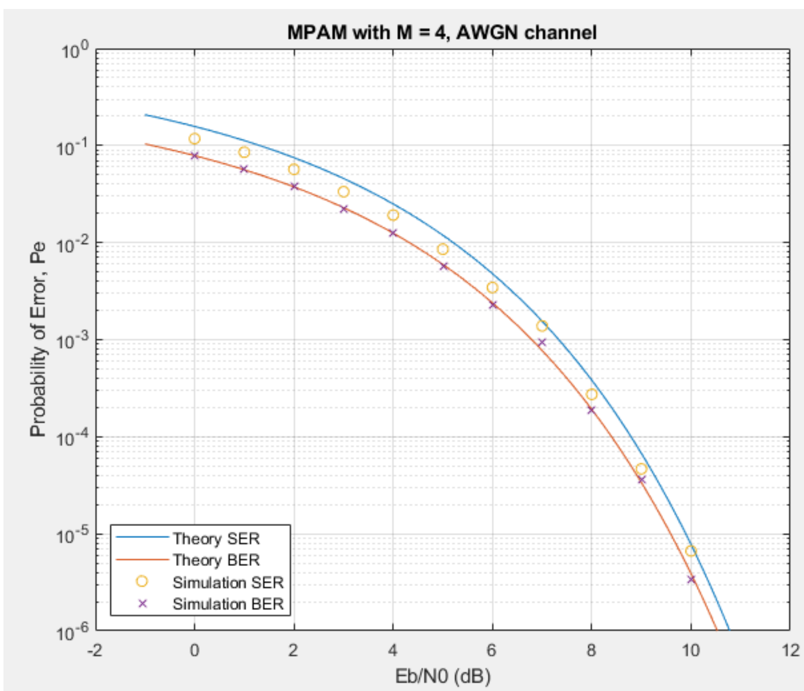

Figure 8

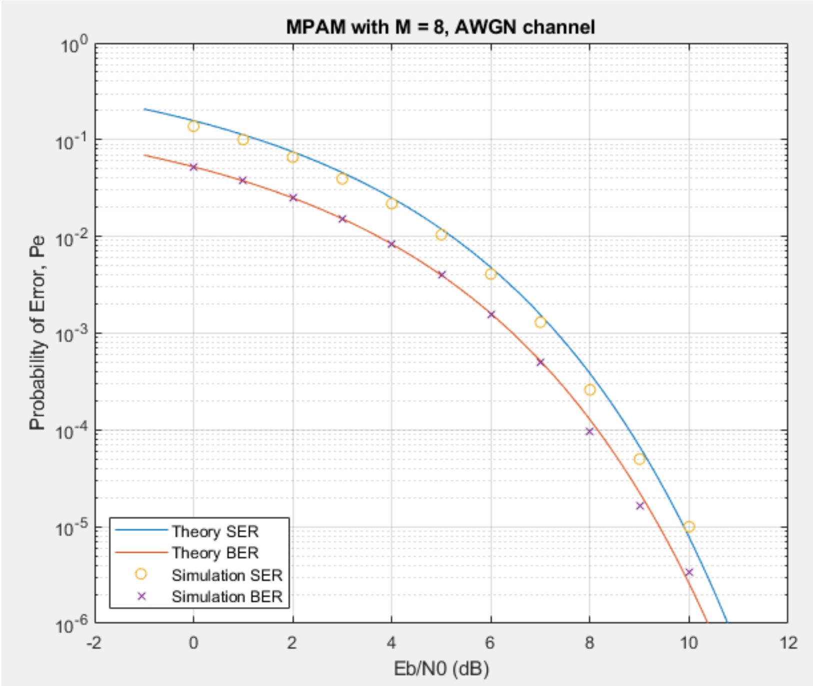

Figure 9

Similar to what we have seen with QAM, increasing M does not change the probability of error since the distance is constant. Again, increasing M in such case will increase the power.

Only MPSK modulation has changed when varying M from the previous results. All of these results have met the theoretical expectations.

B. Fading Channel

MPSK with M = 2, 4, 8.

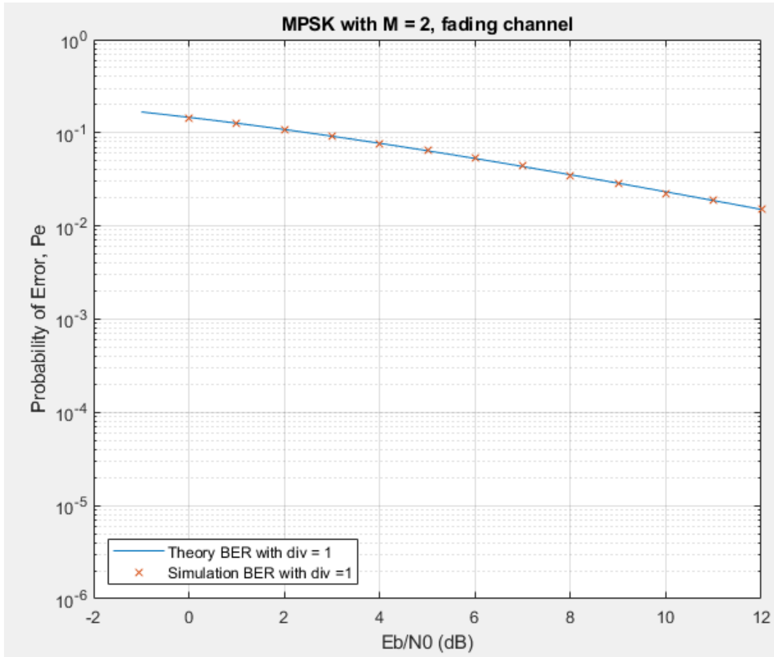

Figure 10

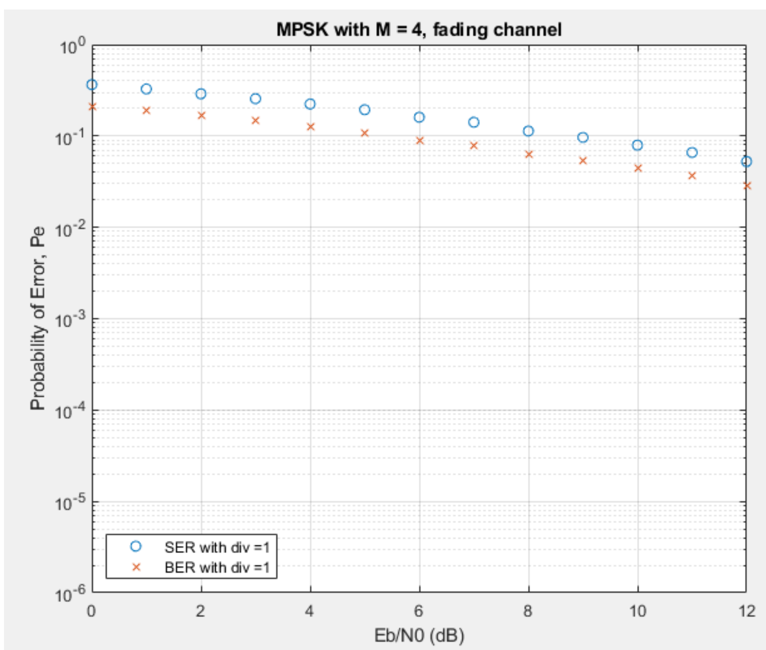

Figure 11

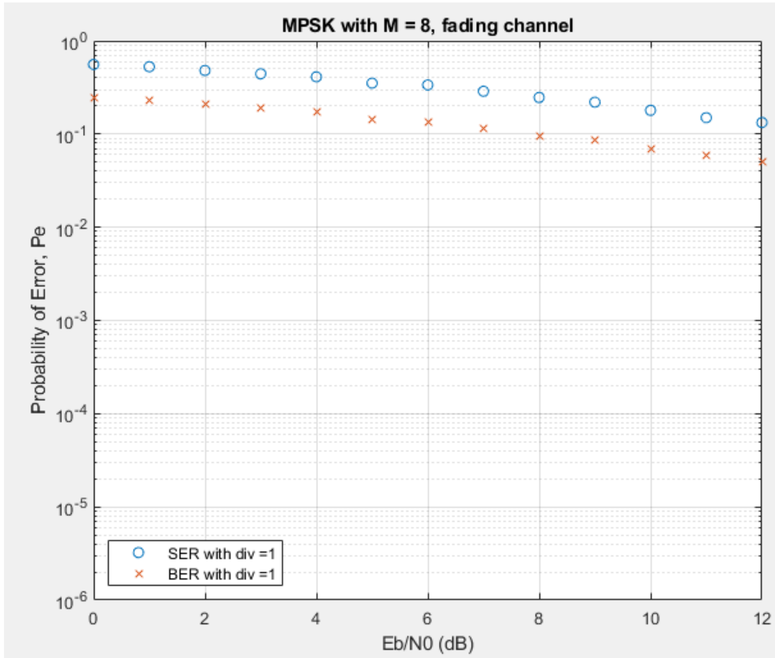

Figure 12

4-QAM modulation

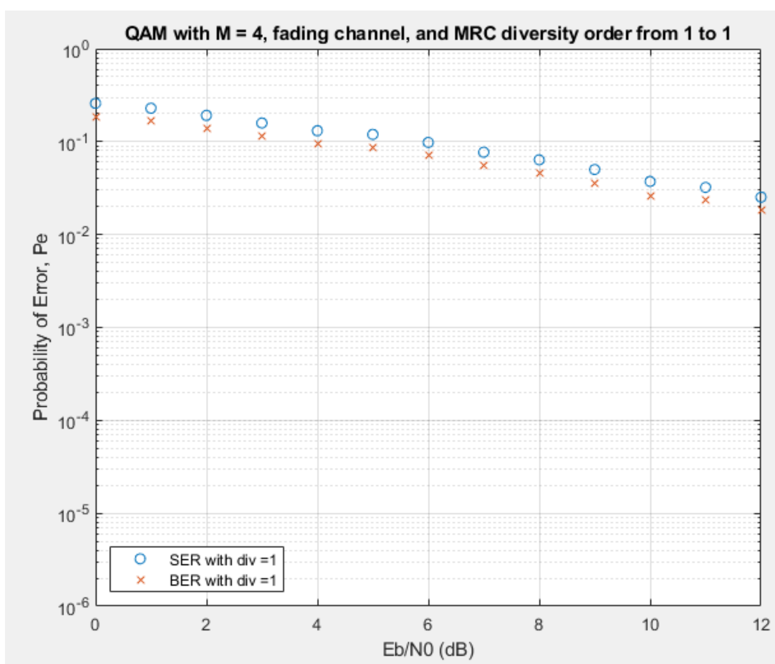

Figure 13

BPAM modulation.

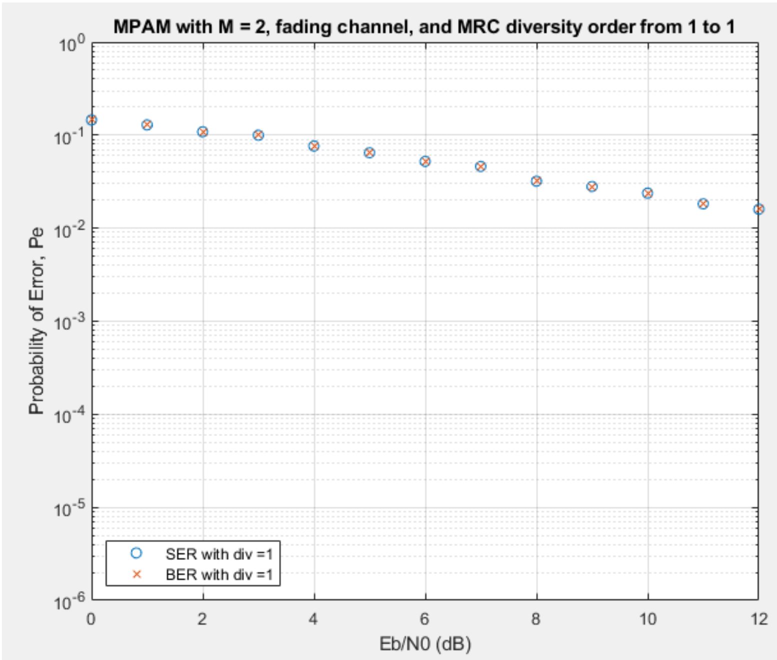

Figure 14

We can see how the effect of fading is severe on the probability of error. Again, increasing M for MPSK increases the error probability but has constant power. We show the theoretical result for BPSK and it agrees with the simulation.

C. Fading Channel with Diversity

We have studied that when we have a fading channel, we can use diversity to enhance the probability of error. We now apply Maximum Ratio Combining (MRC) diversity to the simulation.

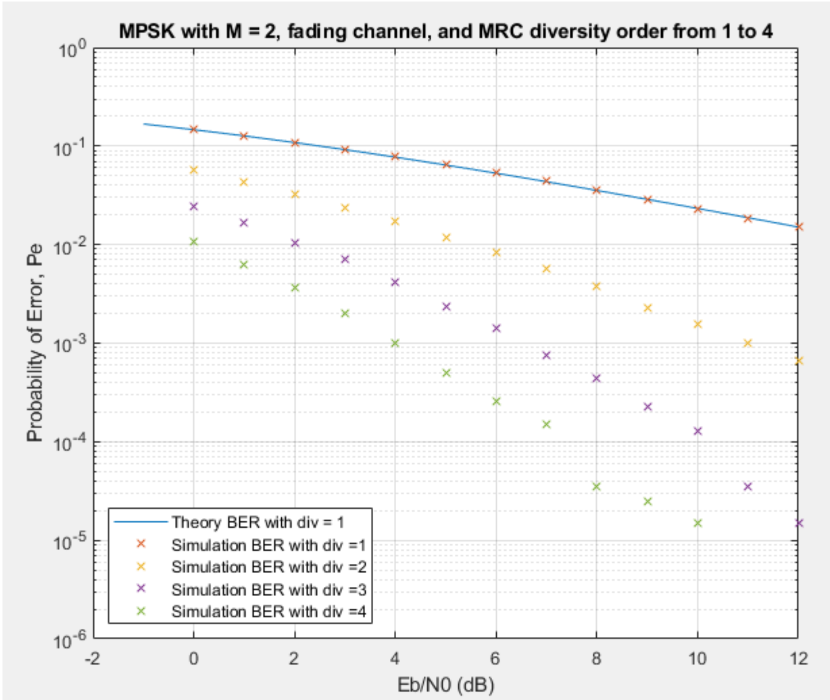

Figure 15

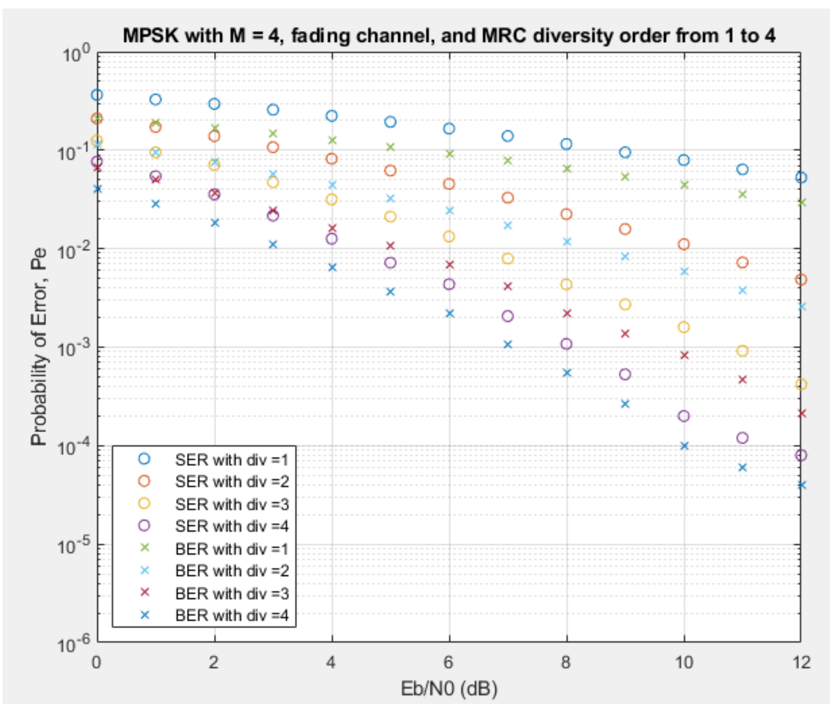

Figure 16

We can see that as the diversity order increases, the error probability decreases by a large margin.

Now, we try and see how Equal Gain Combining (EGC) diversity differs from MRC diversity.

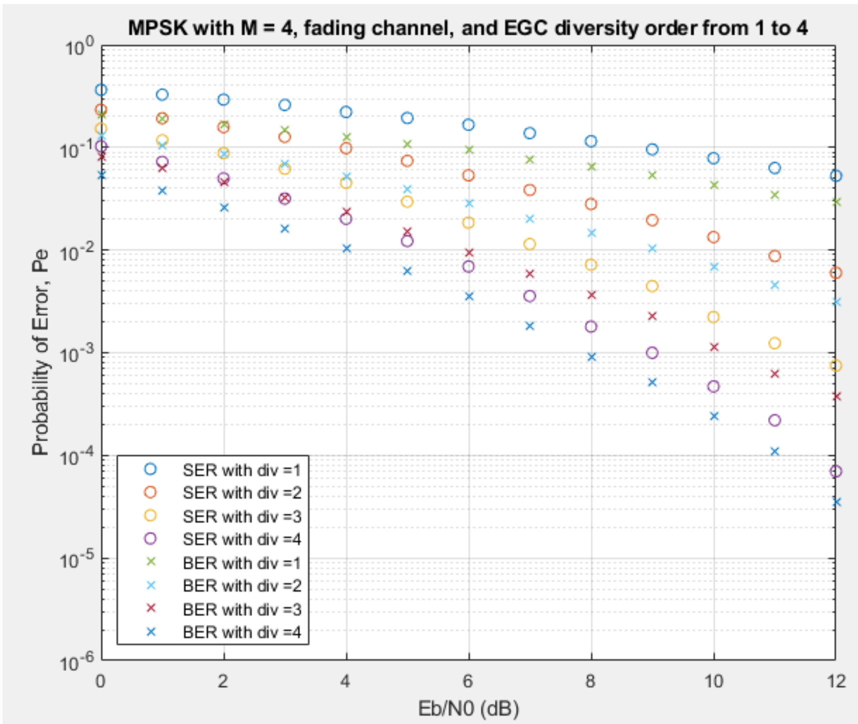

Figure 17

We can see that EGC diversity is not as good as MRC, but the difference between the two is not huge. Both types of diversity give an excellent result in fading channels.

D. Sending Audio Through the System

Now we test recording our voice and sending it through the system. We select QPSK modulation, voice SNR = 10 dB, and sampling at 10 kHz.

The voice over AWGN was audible, but the noise was also audible and had a high pitch, but not very bad. The voice over fading channel was almost inaudible, and the noise level was extremely high. It needs higher SNR to be audible. The audio becomes audible again after applying diversity of factor 4.

V. Conclusion

In conclusion, we have simulated a wireless communication system over AWGN and fading channels. We have written a code that simulates MPSK, QAM, and MPAM for any M in general. We have seen the results of all of these modulations for different levels of m-ary.

Next, we have simulated the system when transmitting in a fading channel. We have seen the effect of fading on the system and how it increases the error probability tremendously.

We then applied diversity to reduce the error in fading channels. We have seen that both MRC and EGC give excellent results in fading channels, and we have tested diversity factor up to 4.

Lastly, we have recorded an audio file and transmitted it through the system. We compared between the audio over AWGN channel and over fading channel and found that the latter was almost inaudible.

VI. References:

[1] Rappaport, Theodore S. *Wireless Communications: Principles and Practice*. Upper Saddle River, N.J.: Prentice Hall PTR, 2002.

[2] Goldsmith, A. (2005) *Wireless Communication*. Cambridge: Cambridge University Press.

Appendix A: MATLAB code

```
% EE 419 - 212 - Term Project
% This project is the work of:
% Ahmed Alghamdi (Student A)
% Abdullah AlKathiry (Student B and C)
% Hussain Balhareth (Student C)

clear; clc; clf;
%% Simulation Settings ( only change this part of the code all needed result)
modelation = "MPSK"; % MPSK, QAM, or MPAM
M_modelation = 4;
recording = true; % Takes time to process
has_fading = true;
diversity_order = 4;% should be 1 if fading is not applied
div_type = 'MRC' ; % MRC or EGC
SNR_voice = 11;% if recording true, choose the SNR you want to hear voice at
%% producing the bits sequence
if recording
    fs=8000; %Sampling frequency
    recObj = audiorecorder(fs,24,1);
    disp('Start speaking.');
    recordblocking(recObj, 3);
    disp('End of Recording.');
    play(recObj);% Play back the recording
    Original = getaudiodata(recObj);% Store data in double-precision array.
    % convert to gray code binary quntized to
    signal = int16((2^9)*(1+(Original/max(abs(Original)))));
    x = reshape(dec2gc(signal,11)',1,[]);
    N_of_bits = length(x);
else
    N_of_bits = 1E4 + 128;
    x=round(rand(1,N_of_bits));
end
%% coordinates of the modulation
if modelation == "MPSK"
    bits_per_symbol = log2(M_modelation);
    coordinate = zeros(M_modelation,2);
    phase = 2*pi/M_modelation;
    %Seting up the coordinates of MPSK
    for j = 0:M_modelation-1
        coordinate(j+1,:) = [cos(j*phase),sin(j*phase)];
    end
elseif modelation == "QAM"
    bits_per_symbol = log2(M_modelation);
    %Seting up the coordinates of QAM using the function qammod
    complex = qammod(0:M_modelation-1,M_modelation);
    coordinate = [real(complex'),imag(complex')];

elseif modelation == "MPAM"
    bits_per_symbol = log2(M_modelation);
    coordinate = zeros(M_modelation,2);
    %Seting up the coordinates of MPAM
    for j=1:M_modelation/2
        coordinate(j,:) = [2*j - 1 , 0];
        coordinate(j+ M_modelation/2 ,:) = [-2*j + 1 , 0];
    end
end
%% bits to symbols
w = rem(N_of_bits,bits_per_symbol); %Remainder
w1 = ceil(N_of_bits/bits_per_symbol);%number of symbols
```

```
if w~=0
    %We remove the extra bits if there are any
    x = [x,zeros(1,bits_per_symbol -w)];
end
N_of_bits = length(x);
y = zeros(w1,bits_per_symbol); %Setting up the symbols matrix
% bit gray coded to symbol
for j = 1:w1
    y(j,:)=x((j-1)*bits_per_symbol+1:j*bits_per_symbol);
end
s_num = gc2dec(y)+1;
coordinate_s =zeros(length(s_num),2);
for j = 1 : length(s_num)
    coordinate_s(j,:) = round(coordinate(s_num(j),:));
end
%% AWGN, Fading, and Diversity
Counter = 1;
for EbN0dB=1:12 %Vary SNR from 0 dB to 12dB
    EbN0=10^(EbN0dB/10); %SNR in linear
    N0=1/EbN0; %Calculate N0
    if has_fading
        %adding fading
        fading = sqrt(0.5)*randn([size(coordinate_s),diversity_order]);
        fading = sqrt(fading(:,1,:).^2 + fading(:,2,:).^2);
    else
        %not adding fading by making fading = 1
        fading = ones([size(coordinate_s),diversity_order]);
    end
    coordinate_f = fading.*coordinate_s; %Symbols on coordinate after fading
    %Adding AWGN
    noise = sqrt(N0/2)*randn(size(coordinate_f));
    coordinate_r = coordinate_f + noise; %Symbols on coordinate after fading + AWGN
    %Applying diversity
    if div_type == 'MRC'
        coordinate_r = cumsum(fading.*coordinate_r,3); %MRC
    elseif div_type == 'EGC'
        coordinate_r = cumsum(coordinate_r,3); %EGC
    end
    r_recived = zeros(length(s_num),diversity_order); %Matrix set  up
    %Calculating for multiple orders of diversity
    for i =1:diversity_order
        for j =1:length(s_num)
            [M,I]= min(sum((coordinate-coordinate_r(j,:,i)).^2,2));
            r_recived(j,i) = I ;
        end
    %Converting the symbols back to a bit stram
    bit_recived(i,:) = reshape(dec2gc(r_recived(:,i)-1,bits_per_symbol)',1,[]);
    end
    % Calculating SER and BER
    symbol_er(Counter,:) = sum((r_recived-s_num)~=0)/length(s_num); %SER
    bit_er(Counter,:) = sum((bit_recived-x)~=0,2)/N_of_bits; %BER
    EbN0dB_s(Counter)=EbN0dB; %The x-axis in the plot
    if recording && Counter == SNR_voice
       % To play the audio later on
       voice_recived = bit_recived(diversity_order,:);
    end
    clc;
    Counter=Counter+1
end
%% Plotting
clf
EbN0dB = -1:0.1:12; %logarithmic
```

```
EbN0=10.^(EbN0dB/10); %linear
if ~has_fading
    % Theoritical plots for AWGN channel
    d = sqrt(sum((coordinate(1,:)-coordinate(2,:)).^2));
    if modelation == "MPSK"
        Pe_= erfc(0.5*d*sqrt(EbN0));
        Pe_BER = Pe_/bits_per_symbol;
    elseif modelation == "QAM"
        Pe_= 2*(1-1/sqrt(M_modelation))*erfc(sqrt(EbN0));
    else
        Pe_= erfc(0.5*d*sqrt(EbN0));
        Pe_BER = Pe_/bits_per_symbol;
    end
    if M_modelation == 2
        semilogy(EbN0dB,Pe_BER,EbN0dB_s,bit_er,'x')
        legend('Theory BER','Simulation BER','Location','southwest');
    elseif modelation == "QAM"
        semilogy(EbN0dB,Pe_,EbN0dB_s,symbol_er,'o',EbN0dB_s,bit_er,'x')
        legend('Theory SER','Simulation SER','Simulation BER','Location','southwest');
    else
        semilogy(EbN0dB,Pe_,EbN0dB,Pe_BER,EbN0dB_s,symbol_er,'o',EbN0dB_s,bit_er,'x')
        legend('Theory SER','Theory BER','Simulation SER','Simulation
BER','Location','southwest');
    end
    title(modelation + " with M = " +M_modelation + ", AWGN channel");
elseif M_modelation == 2 && modelation ~= "MPAM"
    % Theoritical plot of BER over fading channel for BPSK only
    Pe_fading = 0.5*(1-sqrt(EbN0./(1+EbN0)));
    semilogy(EbN0dB,Pe_fading); hold on; %Theory BER
    semilogy(EbN0dB_s,symbol_er,'x');%Sim BER
    legend(["Theory BER with div = 1" "Simulation BER with div ="+
[1:diversity_order]],'Location','southwest');
    title(modelation + " with M = " +M_modelation + ", fading channel, and "+ div_type
+" diversity order from 1 to " + diversity_order);
else
    semilogy(EbN0dB_s,symbol_er,'o',EbN0dB_s,bit_er,'x')
    %legend("Fading SER", "Fading BER");
    legend(["SER with div ="+ [1:diversity_order] , "BER with div ="+
[1:diversity_order]],'Location','southwest');
    title(modelation + " with M = " +M_modelation + ", fading channel, and "+ div_type
+" diversity order from 1 to " + diversity_order);
end
ylim([1E-6 1]); grid on;
xlabel("Eb/N0 (dB)"); ylabel("Probability of Error, Pe");
%% Play the recording
if recording
    signal_recoverd = gc2dec(reshape(voice_recived,11,[])');
    signal_recoverd1 = (signal_recoverd/(2^9) - 1)*max(abs(Original));
    soundsc(signal_recoverd,fs)
end
```