PERFORMANCE OF QPSK SYTEM IN AWGN AND RAYLEIGH FADING WITH AND WITHOUT INTERLEAVER

MAMALESHWAR KOVI GOWRI KUMAR (40126008), MITALI VERMA (40119650), RIYA YADAV (40127639)

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING, CONCORDIA UNIVERSITY

MONTREAL, QUEBEC, CANADA

INTRODUCTION

In Wireless Communications, fading is the variation of the attenuation of a signal with different variables. These variables include time, geographical position, and radio frequency. Fading is often modeled as a random process. In wireless systems, fading may either be due to multipath propagation, referred to as multipath-induced fading, weather, or shadowing from obstacles affecting the wave propagation, sometimes referred to as shadowing [1].

The presence of reflectors in the environment surrounding a transmitter and receiver create multiple paths that a transmitted signal can traverse. As a result, the receiver sees the superposition of multiple copies of the transmitted signal, each traversing a different path. Each signal copy will experience differences in attenuation, delay and phase shift while travelling from the source to the receiver. This can result in either constructive or destructive interference, amplifying or attenuating the signal power seen at the receiver. Strong destructive interference is frequently referred to as a deep fade and may result in temporary failure of communication due to a severe drop in the channel signal-to-noise ratio [1].

We generally encounter the following types of fading: Slow fading, Fast fading, Flat fading and Frequency selective fading. In cellular and personal communication systems the fading encountered is generally slow. This is because receiver velocities are limited to normal vehicle speeds. It is important to remember that flat/frequency-selective and slow/fast fading are independent phenomena. A receiver may experience any combination: flat and slow, flat and fast, frequency-selective and slow, and frequency-selective and fast fading [2].

To improve the performance of the system under fading, an interleaver is used. It can be seen that error correcting code alone does not improve the performance of the system, but in fact degrades it. The introduction of the interleaver leads to improvement in the BER of the system.

1. COMMUNICATION SYSTEM

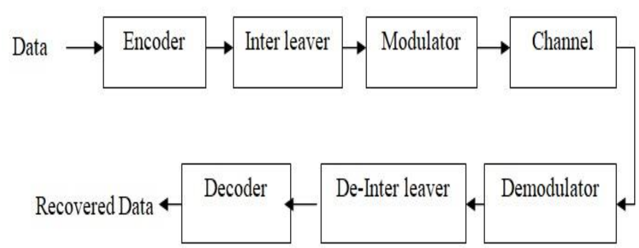

Fig.1. Block diagram of the communication system

The Transmitter of the communication System makes use of QPSK mapping, and is encoded with the (15,11) error correcting Hamming Code, as discussed in Part A of the project. Part A of the project consisted of the performance of the system under AWGN and did not consider fading. It was seen that the performance of the system improved with the introduction of the (15,11) Hamming code. Now, we consider two additional blocks in the communication system, namely interleaver and de-interleaver.

2. CHANNEL

We consider a channel with Additive White Gaussian Noise (AWGN) in the presence of fading. Further, there is an effect of doppler spread due to mobile movement in the channel. Additive white Gaussian noise is additive, i.e., the received signal is equal to the transmitted signal plus noise. Just like the white color which is composed of all frequencies in the visible spectrum, white noise refers to the idea that it has uniform power across the whole frequency band. As a consequence, the Power spectral density of white noise is constant for all frequencies. Further, the probability distribution of the noise is Gaussian, with zero mean [3].

In our scenario, the bit rate = 1Mbps, and the mobile is moving at a speed of 60km/Hr. The carrier frequency is 10Ghz. Also, we consider the case of Flat fading.

To determine whether the type of fading is fast or slow, the following calculations are made: Information rate is 1 Mb/s, carrier frequency is 10GHz and the speed is 60 Km/hr.

$T_{sig} = 2/R$, where R is the information rate.

Therefore, $T_{sig} = 2/ (1 \times 10^6) = 2\ \mu s$

$Tcoh = 9/16\pi F_d$

$F_d = V/C= (16.667 \times 10 \times 10^9) / (3 \times 10^8)$

⇨ 555.57

$Tcoh = 9\ (16 \times \pi \times 555.57) = 322.4\ \mu s$

Where

Tsig = Signal duration

Tcoh = Coherence time

Fd = Doppler Frequency

V = Velocity in m/s

Since, $T_{coh} > T_{sig}$ the fading is Slow Fading. Hence, the fading model to be considered is Rayleigh fading, as the fading is calculated to be slow and flat.

To determine the number of symbols affected by the same fading coefficient, we perform the following calculation:

No. of Symbols = T_{coh} / T_{sig} = (322.4 μs)/ (2 μs) = 161.

161 symbols are affected by the same fading coefficient.

3. INTERLEAVER

Interleaving refers to the process of re-shaping the transmitted bits to reduce the effect of burst errors. In communication systems, errors usually occur in bursts. A burst error is a continuous sequence of symbols, received over a communication channel, such that the first and last symbols are in error and there exists no continuous subsequence of correctly received symbols within the error burst [4]. Consequently, due to the presence of burst errors, when the number of errors in the received codeword exceeds the error correcting capacity of the error correcting code, the transmitted message is not recovered properly, hence leading to errors. The interleaver aims at spreading the error bursts, in order to make the error distribution more uniform.

The de-interleaver does the reverse process of the interleaver. The de-interleaver block is present after demodulation. After the data passes this block, it is sent to the decoder for error detection and correction.

The depth of an interleaver is defined as the minimum separation in symbol periods at the output of the interleaver between any two symbols that were adjacent at the input of the interleaver [5]. It is calculated as follows:

Depth = Tcoh/Tsig, where Tsig = (11/15) * Tsig

= (11/15) x 2 μs = 1.467 μs

Depth = (322 μs)/ (1.467 μs) = 220

Therefore, the depth is calculated to be 220.

4. SIMULATION RESULTS

4.1 PERFORMANCE OF UNCODED QPSK UNDER FADING

Continuing from Part A, the code was written for the simulation of the BER of QPSK under Rayleigh fading. It was observed that the performance of the system was degraded when the channel was considered to have Fading. For verification, the theoretical value of BER under fading was plotted alongside the simulated curve. It was seen that the curves matched, hence proving our result was correct. The theoretical value of BER under fading is given by the expression:

BER= 0.5(1-sqrt((Eb/No)/(1+(Eb/No)))

Fig.2 shows the curves for the scenario discussed above

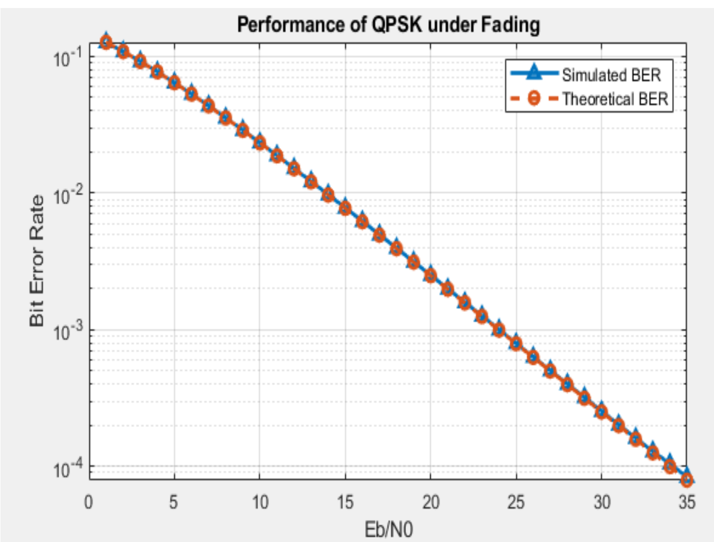

Fig.2. Performance of uncoded QPSK under fading

4.2 PERFORMANCE OF CODED QPSK UNDER FADING

The (15,11) Error Correcting Hamming Code was used to encode the data sequence. We observed that the performance of the system improved with coding when fading was not present. However, the performance of the system was degraded with the introduction of fading. As discussed earlier, this is due to the fact that fading introduces burst errors, which gives us no continuous subsequence of correctly received symbols within the error burst. As a result, the number of errors in the received sequence exceeds the error correcting capability of the Hamming code. This results in a poorer BER.

The curve for simulated BER of coded QPSK under fading was compared with the simulated BER of uncoded QPSK under fading.

Fig.3 shows the curves for the above-mentioned scenario

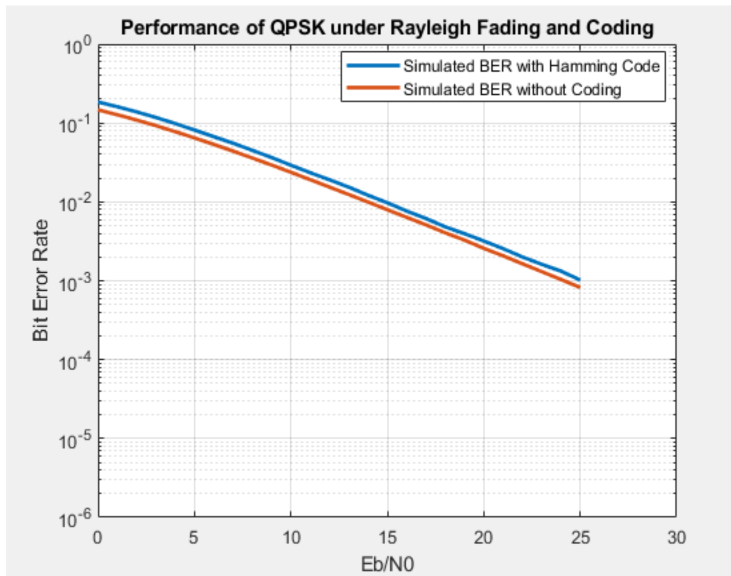

Fig.3. Performance of coded QPSK under Rayleigh fading

4.3 PERFORMANCE OF CODED QPSK UNDER FADING WITH INTERLEAVER

To reduce the influence of burst errors, the interleaver is used. As discussed earlier, the interleaver attempts to spread the burst errors, in order to make the error distribution more uniform. It can be implied that the interleaver distributed the effect of the deep fade, hence enabling the error correcting code to correct most of the erroneous received codewords. It can be seen that the performance of the system improves significantly with the introduction of the interleaver.

The MATLAB function “reshape” was used to perform the interleaving process. The curve was compared to the previous simulation results. It was seen that the curves for the BER of QPSK with interleaver crossed the curve for BER of uncoded QPSK around the 7dB mark. The graph of the scenario is as follows:

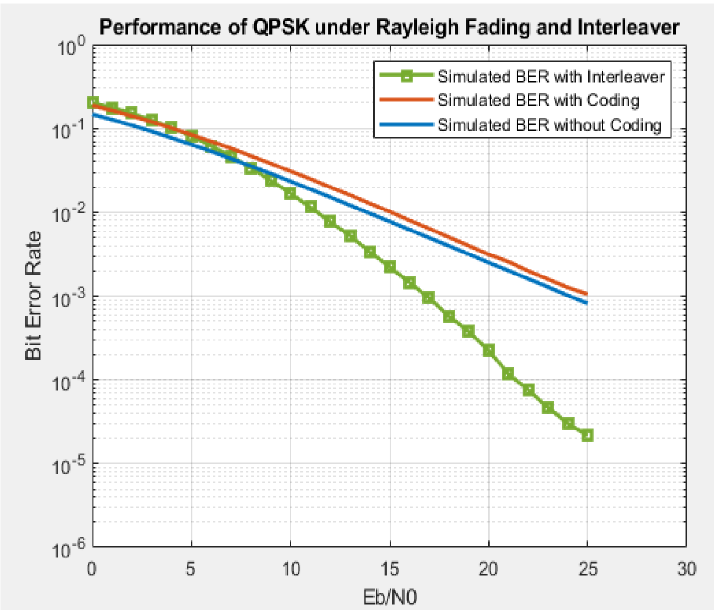

Fig.4. Performance of coded QPSK under Rayleigh fading and interleaver

5. REFERENCES

[1]. https://en.wikipedia.org/wiki/Fading

[2]. Bruce A. Black et al, “Introduction to Wireless Systems”, Prentice Hall, 2008

[3].https://wirelesspi.com/additive-white-gaussian-noise-awgn/

[4].https://en.wikipedia.org/wiki/Burst_error-correcting_code#Interleaved_codes

[5].https://web.stanford.edu/group/cioffi/doc/book/chap11.pdf

APPENDIX

The codes which were used to analyse the performance of the QPSK system are given below.

1. UNCODED QPSK UNDER AWGN AND FADING

```
clc
clear all
%Defining Input Size
input=161*10^6;
x = rand(1,input)>0.5;
j=1;

%Modulation of Data using QPSK
for i = 1:2:length(x)
    if x(i)==0 && x(i+1) == 0
        s(j) = -1-1*1i;
    elseif x(i)==0 && x(i+1) == 1
        s(j)= -1+1*1i;
    elseif x(i)==1 && x(i+1) == 0
        s(j)= 1-1*1i;
    else
        s(j)= 1+1*1i;
    end
    j=j+1;
end

%Noise Generation for Fading Channel
 h=sqrt(1/2)*(randn(1,input/(2*161))+1i*randn(1,input/(2*161)));
 hprime= abs(h);
 for ii= 0:(input/(2*161))-1
     faded_sig(161*ii+1:(ii+1)*161)=s(161*ii+1:(ii+1)*161)*hprime(ii+1);
 end

for SNRdb = 1:35
    No = 10^(-SNRdb/10);
    %Noise Generation for AWGN Channel
    noise = ((No/2)^0.5)* (randn(1,(input)/2) + 1i * randn(1,(input)/2));
    receivedsig = faded_sig + noise;
    for kk= 0:(length(receivedsig)/161)-1
        receivedsig1(161*kk+1:(kk+1)*161) =
receivedsig(161*kk+1:(kk+1)*161)/hprime(kk+1);
    end
    %Demodulation of Data
    j=1;
    for i = 1:length(receivedsig1)
        if real(receivedsig1(i))>0 && imag(receivedsig1(i))>0
           y(j) = 1;
           y(j+1)=1;
        elseif real(receivedsig1(i))>0 && imag(receivedsig1(i))<0
           y(j) = 1;
           y(j+1)=0;
        elseif real(receivedsig1(i))<0 && imag(receivedsig1(i))>0
           y(j) = 0;
           y(j+1)=1;
        else
           y(j) = 0;
           y(j+1)=0;
```

```
        end
    j = j+2;
    end

    error=0;
    for i=1:length(x)
        if y(i) ~= x(i)
           error= error+1;
        end
    end
%Calculation of Simulated Probability of Error
Pe(SNRdb)= error/length(x);
end

%Calculation of Theoretical Probability of error in presence of Fading
SNRdb = 1:35;
SNR=10.^(SNRdb/10);
BER(SNRdb)=0.5.*(1-sqrt(SNR./(1+SNR)));

%Plotting the Graphs
semilogy(Pe)
hold on
semilogy(BER)
grid on
xlabel('Eb/N0')
ylabel('Bit Error Rate')
title('Performance of QPSK under Fading')
```

2. CODED QPSK UNDER AWGN AND RAYLEIGH FADING

```
clc
clear all
%Defining input size
input=11*22*10^4;
n=15;
k=11;
SNRdbVec=0:25;

p = [1 1 1 1
    0 1 1 1
    1 0 1 1
    1 1 0 1
    1 1 1 0
    0 0 1 1
    0 1 0 1
    0 1 1 0
    1 0 1 0
    1 0 0 1
    1 1 0 0];
%Generator Matrix
G = [p eye(k)];
H=[eye(n-k) p.'];
%Parity Check Matrix
Ht=H.';
e=[zeros(1,n);diag(ones(1,n))];
%Syndrome Table
```

```
synd = [mod(e*Ht,2) e ];
x = rand(1,input)>0.5;
L=zeros(1,(input*n)/k);

for ii=0:(input/11)-1
    %Encoding the data
    L(15*ii+1:(ii+1)*15) = mod(x(ii*11+1:(ii+1)*11)*G,2);
end
j=1;

%Modulation of data using QPSK
for i = 1:2:length(L)
    if L(i)==0 && L(i+1) == 0
        s(j) = -1-1*1i;
    elseif L(i)==0 && L(i+1) == 1
        s(j)= -1+1*1i;
    elseif L(i)==1 && L(i+1) == 0
        s(j)= 1-1*1i;
    else
        s(j)= 1+1*1i;
    end
    j=j+1;
end
%Generation of noise for Fading channel
     h=sqrt(1/2)*(randn(1,length(s)/(220))+1i*randn(1,length(s)/(220)));
         hprime= abs(h);
for ii= 0:(length(s)/(220))-1
   faded_sig(220*ii+1:(ii+1)*220)=s(220*ii+1:(ii+1)*220)*hprime(ii+1);
end

indexBER = 1;
for SNRdb = SNRdbVec
    No = (10^(-SNRdb/10))*(15/11);
    %Generation of Additive White Gaussian Noise
    noise = ((No/2)^0.5)* (randn(1,(length(s))) + 1i *
randn(1,(length(s))));
    receivedsig = faded_sig + noise;
    for kk= 0:(length(receivedsig)/220)-1
        receivedsig1(220*kk+1:(kk+1)*220) =
receivedsig(220*kk+1:(kk+1)*220)/hprime(kk+1);
    end
    j=1;
    %Demodulation of Data
    for i = 1:length(receivedsig1)
        if real(receivedsig1(i))>0 && imag(receivedsig1(i))>0
           y(j) = 1;
           y(j+1)=1;
        elseif real(receivedsig1(i))>0 && imag(receivedsig1(i))<0
           y(j) = 1;
           y(j+1)=0;
        elseif real(receivedsig1(i))<0 && imag(receivedsig1(i))>0
           y(j) = 0;
           y(j+1)=1;
        else
           y(j) = 0;
           y(j+1)=0;
        end
        j = j+2;
    end
```

```
    c=1;
    for kk=1:n:length(y)-(n-1)
     %Calculation of Syndrome Vector
     S= mod(y(kk:kk+(n-1))*Ht,2);
     %Checking Syndrome table for the corresponding error pattern
         for iii=1:n+1
             if S== synd(iii,1:4)
                 ed=synd(iii,5:19);
             end
         end

         corrected_bits=xor(y(kk:kk+(n-1)),ed);
         decoded_bits(c:c+(k-1))=corrected_bits(5:15);
         c=c+k;

    end
    recv_bits=decoded_bits;
    errors=find(xor(recv_bits,x));
    errors=size(errors,2);
    %Calculation of Bit Error Rate
    BER(indexBER)=errors/input;
    indexBER = indexBER+1;

end
%Plotting the Graph
semilogy(SNRdbVec, BER)
hold on
axis([0 30 10^-6 1.0])
xlabel('Eb/N0')
ylabel('Bit Error Rate')
title('Performance of QPSK under Rayleigh Fading and Coding')
grid on
```

3. CODED QPSK UNDER AWGN AND FADING WITH INTERLEAVER

```
clc
clear all
%Defining input size
input=11*22*10^4;
n=15;
k=11;
SNRdbVec=0:25;

p = [1 1 1 1
    0 1 1 1
    1 0 1 1
    1 1 0 1
    1 1 1 0
    0 0 1 1
    0 1 0 1
    0 1 1 0
    1 0 1 0
    1 0 0 1
    1 1 0 0];
%Generator Matrix
G = [p eye(k)];
H=[eye(n-k) p.'];
%Parity Check Matrix
Ht=H.';
```

```
e=[zeros(1,n);diag(ones(1,n))];
%Syndrome Table
synd = [mod(e*Ht,2) e ];
x = rand(1,input)>0.5;
L=zeros(1,(input*n)/k);

for ii=0:(input/11)-1
    %Encoding the data
    L(15*ii+1:(ii+1)*15) = mod(x(ii*11+1:(ii+1)*11)*G,2);
end

%Interleaver
U=reshape(L,[220,15000]);%depth of interleaver = 220
Ut=U';
h=Ut(:);%Conversion to Column Matrix
f=h';
j=1;

%Modulation of data using QPSK
for i = 1:2:length(f)
    if f(i)==0 && f(i+1) == 0
        s(j) = -1-1*1i;
    elseif f(i)==0 && f(i+1) == 1
        s(j)= -1+1*1i;
    elseif f(i)==1 && f(i+1) == 0
        s(j)= 1-1*1i;
    else
        s(j)= 1+1*1i;
    end
    j=j+1;
end
%Generation of noise for Fading channel
h=sqrt(1/2)*(randn(1,length(s)/(220))+1i*randn(1,length(s)/(220)));
hprime= abs(h);
for ii= 0:(length(s)/(220))-1
   faded_sig(220*ii+1:(ii+1)*220)=s(220*ii+1:(ii+1)*220)*hprime(ii+1);
end

indexBER = 1;
for SNRdb = SNRdbVec
    No = (10^(-SNRdb/10))*(15/11);
    %Generation of Additive White Gaussian Noise
    noise = ((No/2)^0.5)* (randn(1,(length(s))) + 1i *
randn(1,(length(s))));
    receivedsig = faded_sig + noise;
    for kk= 0:(length(receivedsig)/220)-1
        receivedsig1(220*kk+1:(kk+1)*220) =
receivedsig(220*kk+1:(kk+1)*220)/hprime(kk+1);
    end
    j=1;
    %Demodulation of Data
    for i = 1:length(receivedsig1)
        if real(receivedsig1(i))>0 && imag(receivedsig1(i))>0
           y(j) = 1;
           y(j+1)=1;
        elseif real(receivedsig1(i))>0 && imag(receivedsig1(i))<0
           y(j) = 1;
           y(j+1)=0;
        elseif real(receivedsig1(i))<0 && imag(receivedsig1(i))>0
           y(j) = 0;
```

```
            y(j+1)=1;
          else
            y(j) = 0;
            y(j+1)=0;
          end
        j = j+2;
    end

%De-Interleaver
q=reshape(y,[15000,220]);
kt=q';
disint=reshape(kt,[1,3300000]);
c=1;

    for kk=1:n:length((disint))-(n-1)
    %Calculation of Syndrome Vector
     S= mod(disint(kk:kk+(n-1))*Ht,2);
     %Checking Syndrome table for the corresponding error pattern
         for iii=1:n+1
             if S== synd(iii,1:4)
                ed=synd(iii,5:19);
             end
         end

         corrected_bits=xor(disint(kk:kk+(n-1)),ed);
         decoded_bits(c:c+(k-1))=corrected_bits(5:15);
         c=c+k;

    end
    recv_bits=decoded_bits;
    errors=find(xor(recv_bits,x));
    errors=size(errors,2);
    %Calculation of Bit Error Rate
    BER(indexBER)=errors/input;
    indexBER = indexBER+1;
end
%Plotting the Graph
semilogy(SNRdbVec, BER)
hold on
axis([0 30 10^-6 1.0])
xlabel('Eb/N0')
ylabel('Bit Error Rate')
grid on
title('Performance of QPSK under Rayleigh Fading and Interleaver')
```