Name: Vratesh Kumar
USN: 1MS21EC131
Section: 7 B
Subject: Wireless Communication IPCC Lab
Code: EC71
Term: 3-October-2024 to 25-January-2025

Contents

Experiment 1

Analyze Bit Error Rate (BER) performance for BPSK signals over AWGN. Compare the results with theoretical results.

Aim

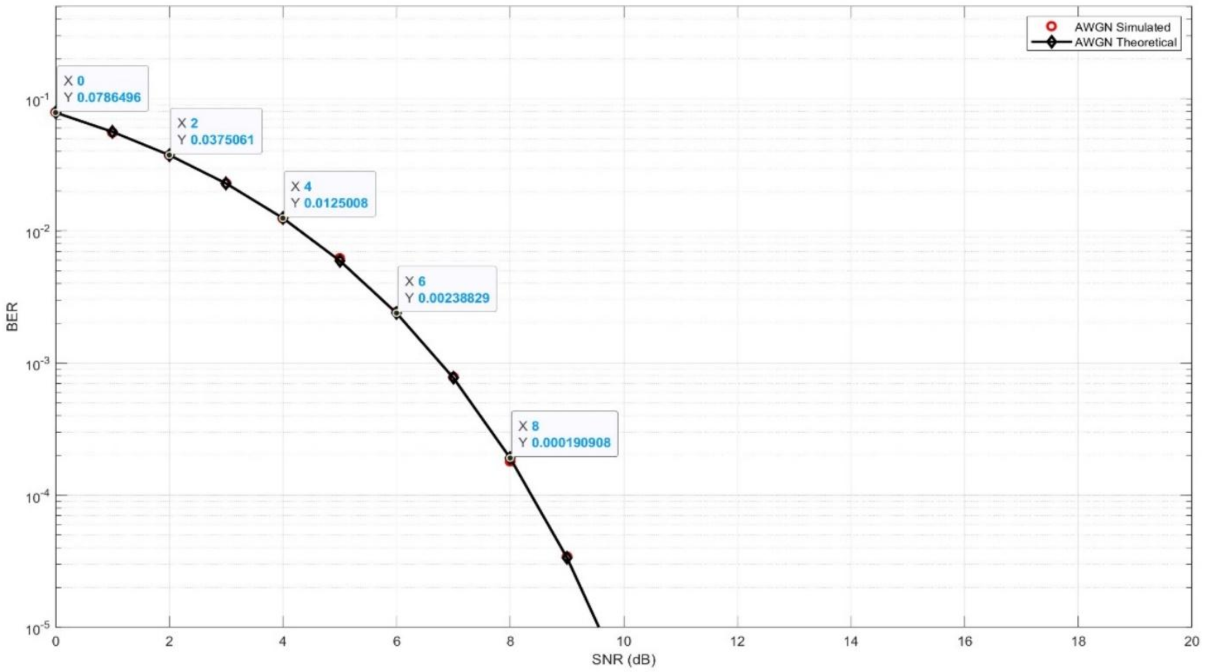
Write a code to compute Bit Error Rate (BER) performance for BPSK signals over AWGN channel. Compare the results with theoretical results**.**

Code

```
clear all;
% Number of information bits
m= 10^5;
%Range of SNR values
snr_dB = 0:1:10;
for j=1:1:length(snr_dB)
   n_err = 0;
   n_bits = 0;
   while n_err < 100
      % Generate sequence of binary bits
      inf_bits=round(rand(1,m));

      % BPSK modulator
      x=2*(inf_bits-0.5);

      % Noise variance
      N0=1/10^(snr_dB(j)/10);

      % Send over Gaussian Link to the receiver
      y=x + sqrt(N0/2)*(randn(1,length(x))+1i*randn(1,length(x)));

      % Decision making at the Receiver
      est_bits=y > 0;

      % Calculate Bit Errors
      diff=inf_bits-est_bits;
      n_err=n_err+sum(abs(diff));
      n_bits=n_bits+length(inf_bits);
   end

   % Calculate Bit Error Rate
   BER(j)=n_err/n_bits;
end

% AWGN Theoretical BER
theoryBerAWGN=0.5*erfc(sqrt(10.^(snr_dB/10)));
semilogy(snr_dB,BER,'or','LineWidth',2);
hold on;
semilogy(snr_dB,theoryBerAWGN,'blad-','LineWidth',2);
legend('AWGN Simulated', 'AWGN Theoretical');
```

```
axis([0 20 10^-5 0.5]);
xlabel('SNR (dB)');
ylabel('BER');
grid on;
```

Output



Inference

The bit error rate (BER) is less to zero in the additive white guassian noise (AWGN) channel for binary phase shift keying (BPSK). It is due to the reason that the SNR is very high and the impact of noise is negligible.

Experiment 2

Analyze Bit Error Rate (BER) performance for BPSK signals over Rayleigh fading channel. Compare the results with theoretical results.

Aim

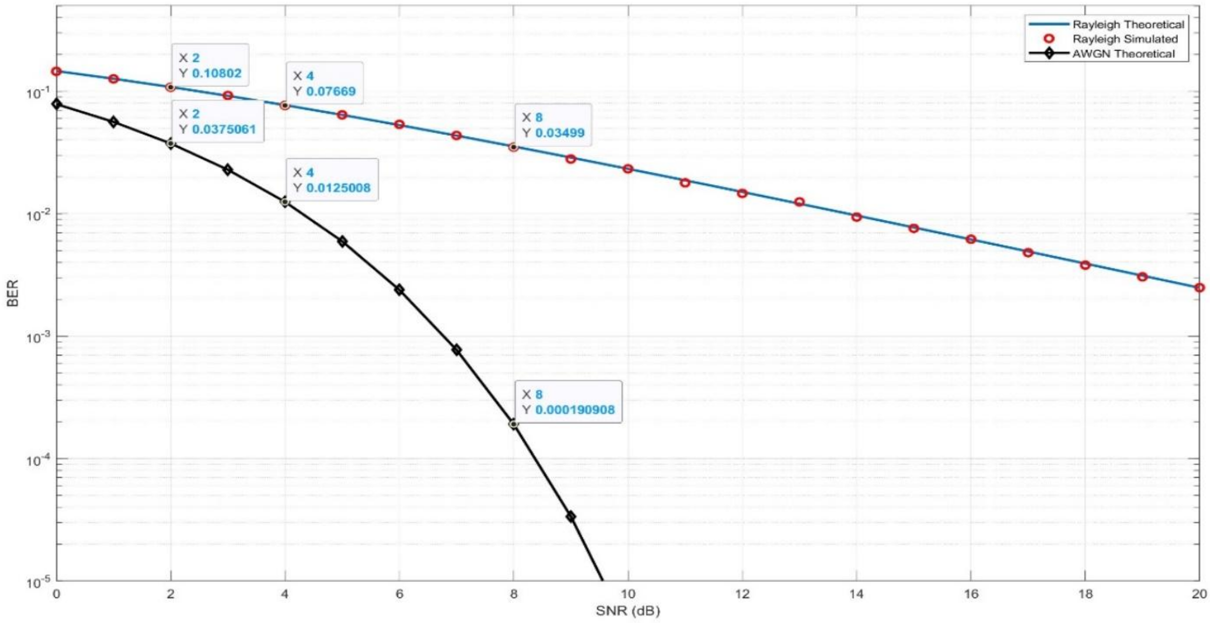
Write a code to compute Bit Error Rate (BER) performance for BPSK signals over AWGN and Rayleigh channel. Compare the results with theoretical results**.**

Code

```
clear all;
% Number of information bits
m= 10^5;
%Range of SNR values
snr_dB = [0:1:20];
for j=1:1:length(snr_dB)
    n_err = 0;
    n_bits = 0;
    while n_err < 100
        % Generate sequence of binary bits
        inf_bits=round(rand(1,m));
        % BPSK modulator
        x=-2*(inf_bits-0.5);
        % Noise variance
        N0=1/10^(snr_dB(j)/10);
        % Rayleigh channel fading
        h=1/sqrt(2)*[randn(1,length(x)) + i*randn(1,length(x))];
        % Send over Gaussian Link to the receiver
        y=h.*x + sqrt(N0/2)*(randn(1,length(x))+i*randn(1,length(x)));
        % decision metric
        y=y./h;
        % Decision making at the Receiver
        est_bits=y < 0;
        % Calculate Bit Errors
        diff=inf_bits-est_bits;
        n_err=n_err+sum(abs(diff));
        n_bits=n_bits+length(inf_bits);
    end
    % Calculate Bit Error Rate
    BER(j)=n_err/n_bits;
end
% Rayleigh Theoretical BER
snr = 10.^(snr_dB/10);
theoryBer=0.5.*(1-sqrt(snr./(snr+1)));
% AWGN Theoretical BER
theoryBerAWGN=0.5*erfc(sqrt(10.^(snr_dB/10)));
semilogy(snr_dB,theoryBer,'-','LineWidth',2);
```

```
hold on;
semilogy(snr_dB,BER,'or','LineWidth',2);
hold on;
semilogy(snr_dB,theoryBerAWGN,'blad-','LineWidth',2);
legend('Rayleigh Theoretical','Rayleigh Simulated', 'AWGN Theoretical');
axis([0 20 10^-5 0.5]);
xlabel('SNR (dB)');
ylabel('BER');
grid on;
```

Output



Inference

Rayleigh fading channels significantly degrade the BER performance of BPSK. This degradation is primarily due to the fading effects, which can cause deep fades that severely attenuate the received signal. As the SNR increases, the BER decreases, but the improvement is less significant compared to AWGN channels.

Experiment 3

Study of Log-Distance path loss propagation model

Aim

a) Write a Matlab program to calculate the path loss for log distance path loss indoor propagation mode.

b) To simulate the log distance path loss model using Matlab.

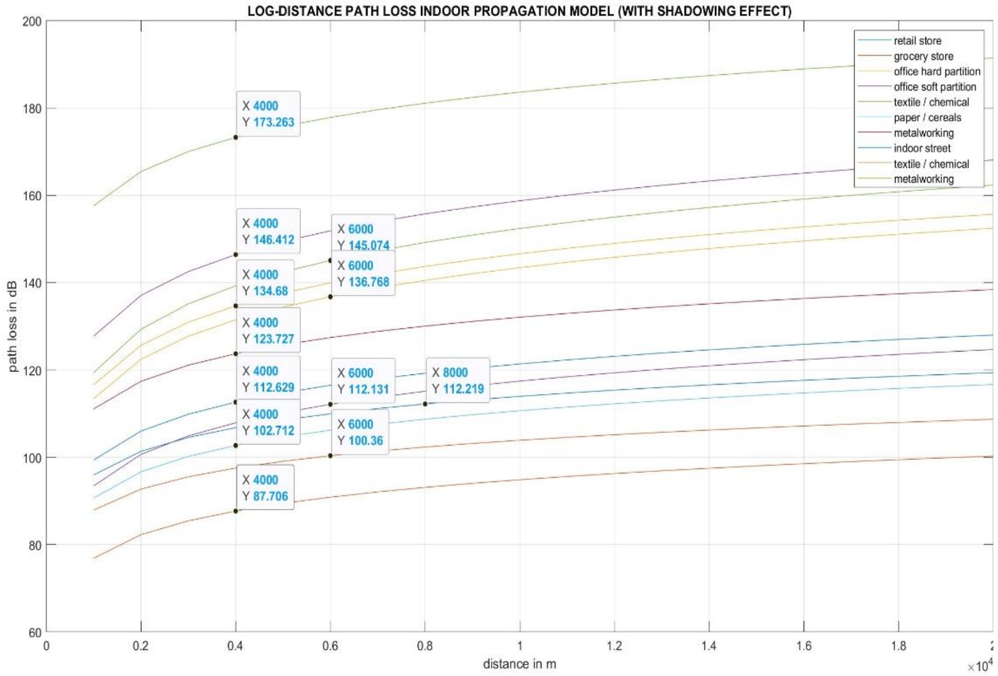
c) To obtain graphical representation by varying various parameters and by considering various terrains.

Log – Distance path loss indoor propagation model with shadowing

Code

```
clc
clear all;
close all;
d0=input('enter the reference distance:');
d=1000:1000:20000;
n= [2.2 1.8 3.0 2.4 2.6 2.0 2.1 1.8 1.6 3.0 3.1 3.3];
f= [914 914 1500 900 1900 1300 4000 1300 1300 900 4000 1300];
sigma= [8.7 5.2 7.0 9.6 14.1 3.0 7.0 6.0 5.8 7.0 9.7 6.8];

for i=1:12
   lambda(i)=3e8/(f(i)*10^6);
   PL_d0(i)=-10*log10((lambda(i)^2)/((4*pi*d0)^2));
   X(i)=sigma(i)*randn(size(PL_d0(i)));
   disp(randn(size(PL_d0(i))));
end
for i=1:12
   for j=1:20
      PL(i,j)=PL_d0(i)+10*n(i)*log10(d(j)/d0)+X(i);
   end
end

%DISTANCE VS PATH LOSS

plot (d,PL);
legend ('retail store', 'grocery store', 'office hard partition', 'office soft partition', 'textile / chemical', 'paper / cereals',
'metalworking', 'indoor street', 'textile / chemical', 'metalworking');
xlabel ('distance in m');
ylabel ('path loss in dB');
title ('LOG-DISTANCE PATH LOSS INDOOR PROPAGATION MODEL (WITH SHADOWING EFFECT');
grid on;
```

Output




```
enter the reference distance:30
    0.5564

   -1.5367

   -0.8031

   -0.4577

   -0.0484

   -1.7721

    0.5364

    0.1012

   -1.1138

   -2.8906

   -0.5757

   -0.1421
```

Inference

A log-normal random variable is used to model the indoor path lass. A shadowing component is incorporated to account for the random variations in signal strength due to obstacles and obstructions in the propagation environment.

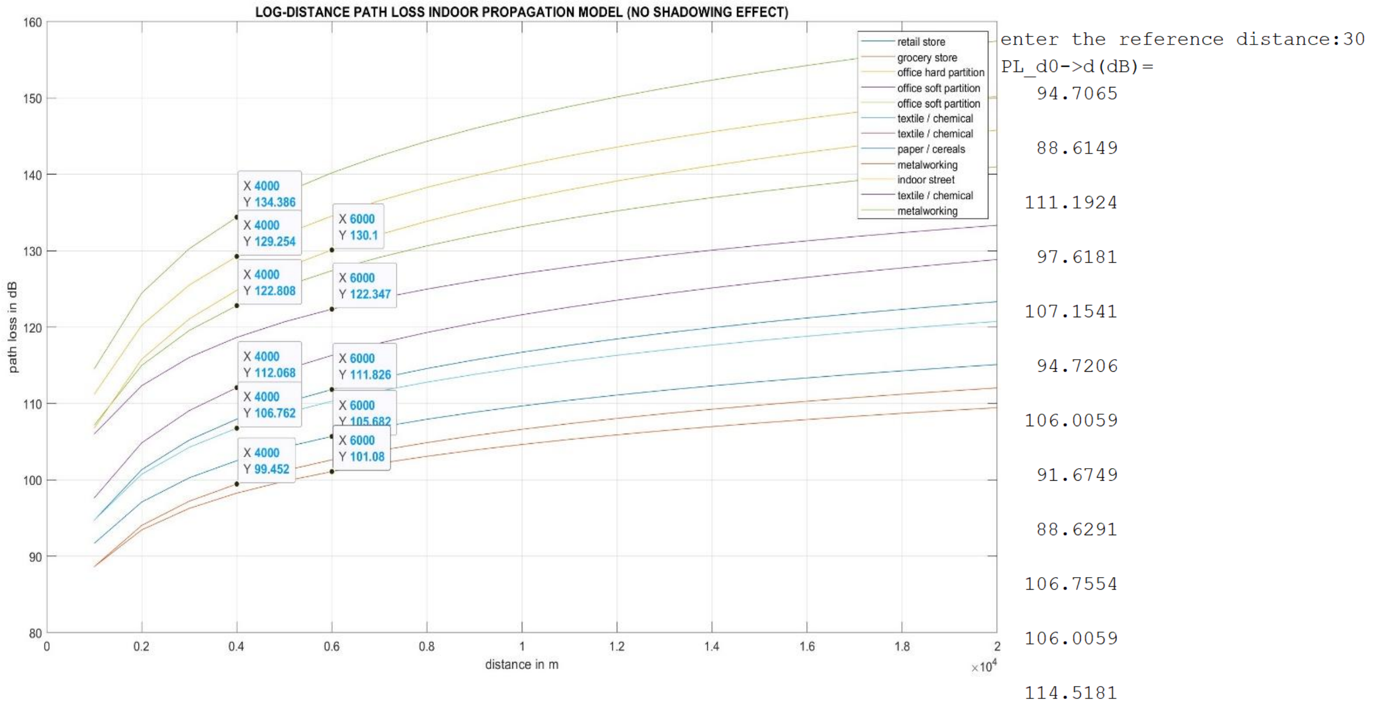
Log – Distance path loss indoor propagation model

Code

```
clc
clear all;
close all;
d0=input('enter the reference distance:');
d=1000:1000:20000;
n= [2.2 1.8 3.0 2.4 2.6 2.0 2.1 1.8 1.6 3.0 2.1 3.3];
f= [914 914 1500 900 1900 1300 4000 1300 1300 900 4000 1300];
X= 0;
for i=1:12
lambda(i)=3e8/(f(i)*10^6);
PL_d0(i)=-10*log10((lambda(i)^2)/((4*pi*d0)^2))+X;
end
disp('PL_d0->d(dB)=');
```

```matlab
for i=1:12
    for j=1:20
        PL(i,j)= PL_d0(i)+10*n(i)*log10(d(j)/d0);
    end
disp(PL(i));
end


%DISTANCE VS PATH LOSS

plot (d, PL);
legend ('retail store', 'grocery store', 'office hard partition', 'office soft partition', 'office soft partition', 'textile / chemical', 'textile / chemical', 'paper / cereals', 'metalworking', 'indoor street', 'textile / chemical', 'metalworking');
xlabel ('distance in m');
ylabel ('path loss in dB');
title ('LOG-DISTANCE PATH LOSS INDOOR PROPAGATION MODEL (NO SHADOWING EFFECT');
grid on;
```

Output



Inference

The log - distance path loss model without shadowing effects, provides a basic estimation of signal attenuation as a function of distance. As there is no shadowing effect the path loss is less compared to log – normal path loss model.

Experiment 4

Study of HATA propagation model.

Aim:

a) Write a matlab program to calculate the pathloss for HATA outdoor propagation model

b) Simulate the Hata path loss model using matlab

c) Obtain the graphical representation by varying various parameters and by considering the various terrains.

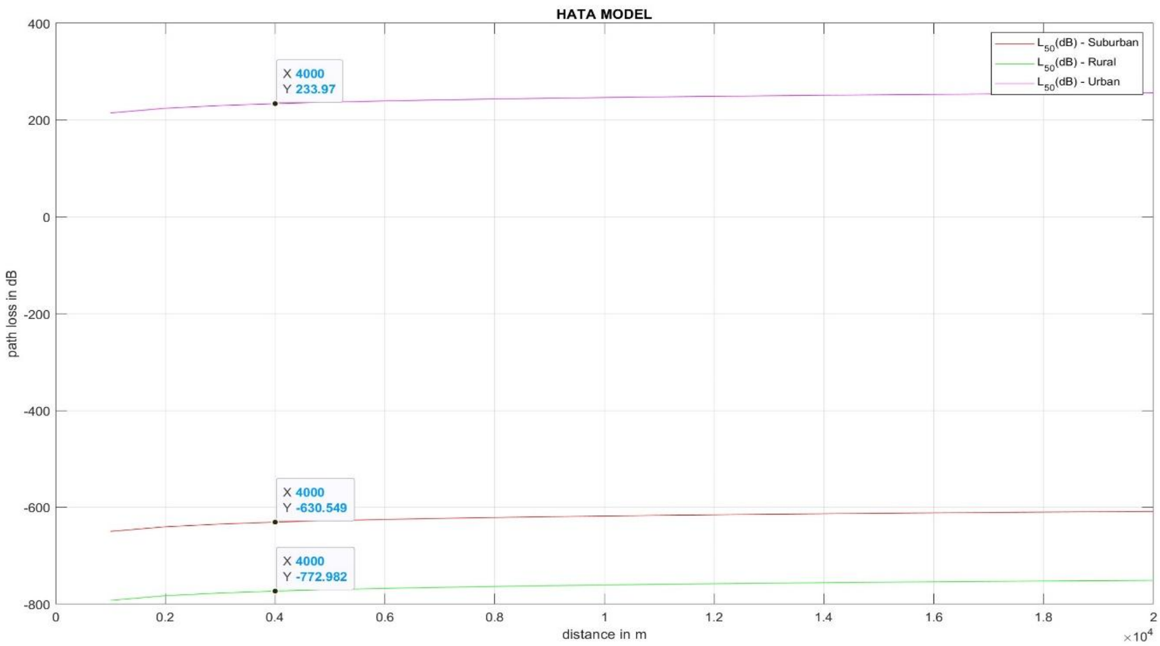
HATA Model for distance varied

Code

```
%HATA Model for distance varied
clc
clear all;
close all;
hte=input('enter the tx height:');
hre=input('enter the rx height');
d=1000:1000:20000;
f=input('enter the frequency:');
% MEDIUM-SMALL CITY (SUB URBAN AREA)
ahre_1=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_1=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_1;
L50_dB_1=L50_1-(2*log10(f/28)*log10(f/28))-5.4;
% MEDIUM SMALL CITY RURAL / OPEN AREA
ahre_2=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_2=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_2;
L50_dB_2=L50_2-(4.78*log10(f)*log10(f))+18.33*log10(f)-40.98;
% LARGE CITY URBAN AND SUB-URBAN AREA
if f <3000000000
   ahre_3=8.29*(log10(1.54*hre))*(log10(1.54*hre))-1.1;
else
   ahre_3=3.2*(log10(11.75*hre))*(log10(11.75*hre))-4.97;
end
L50_3=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_3;
L50_dB_3=L50_3-(2*log10(f/28)*log10(f/28))-5.4;

% Distance vs path loss
figure
plot (d,L50_dB_1,'r',d,L50_dB_2,'g',d,L50_dB_3,'m');
legend ('L_5_0(dB) - Suburban', 'L_5_0(dB) - Rural', 'L_5_0(dB) - Urban');
xlabel ('distance in m');
ylabel ('path loss in dB');
title ('HATA MODEL');
grid on;
```

Output

```
enter the tx height:100
enter the rx height:100
enter the frequency:925000000
```



Inference

The path loss increases logarithmically with distance, as indicated by the {[44.9 - 6.55 $\log_{10}$(hte)] $\log_{10}$(d)} term.

HATA Model Receiver height varied

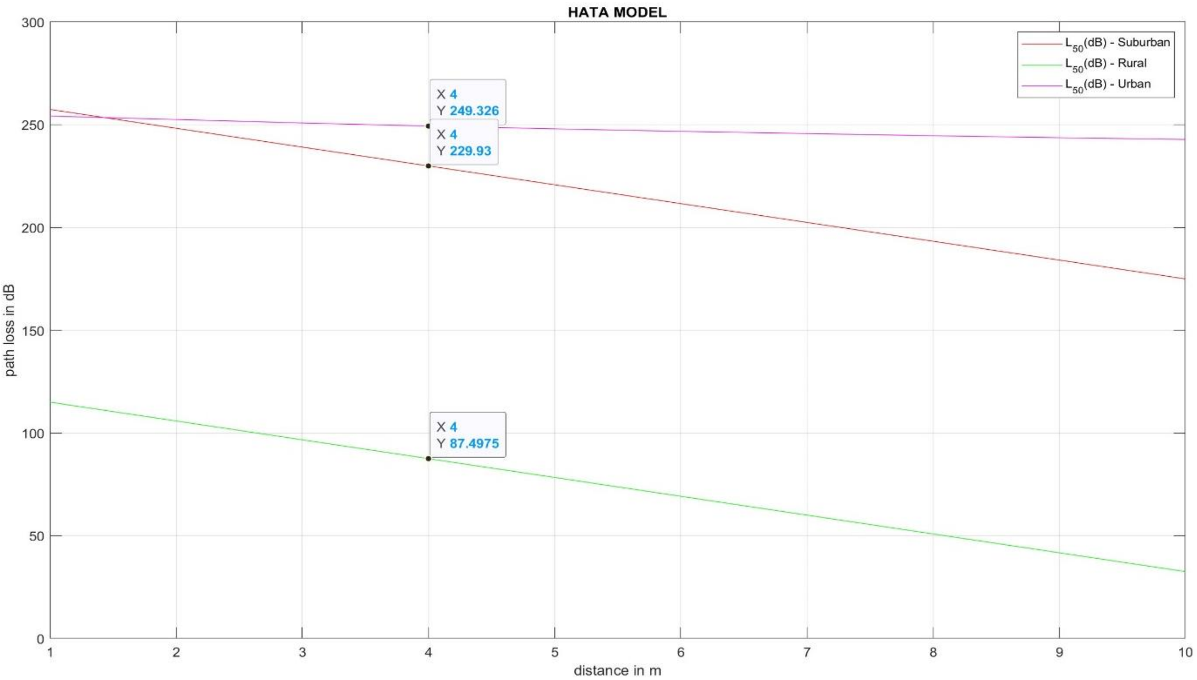
Code

```
%HATA MODEL Receiver height varied
clc
clear all;
close all;
hte = input ('enter the tx height:');
hre = 1:10;
d=input ('enter the distance between tx and rx :');
f= input('enter the frequency:');
```

```
% MEDIUM-SMALL CITY (SUB URBAN AREA)

ahre_1=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_1=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_1;
L50_dB_1=L50_1-(2*log10(f/28)*log10(f/28))-5.4;

% MEDIUM SMALL CITY RURAL / OPEN AREA
ahre_2=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_2=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_2;
L50_dB_2=L50_2-(4.78*log10(f)*log10(f))+18.33*log10(f)-40.98;

% LARGE CITY URBAN AND SUB-URBAN AREA
if f <3000000000
    ahre_3=8.29*(log10(1.54*hre)).*(log10(1.54*hre))-1.1;
else
    ahre_3=3.2*(log10(11.75*hre)).*(log10(11.75*hre))-4.97;
end
L50_3=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_3;
L50_dB_3=L50_3-(2*log10(f/28)*log10(f/28))-5.4;

% rx height vs path loss
figure
plot (hre, L50_dB_1,'r', hre, L50_dB_2,'g',hre, L50_dB_3,'m');
legend ('L_5_0(dB) - Suburban', 'L_5_0(dB) - Rural', 'L_5_0(dB) - Urban');
xlabel ('distance in m');
ylabel ('path loss in dB');
title ('HATA MODEL');
grid on;
```

Output

```
enter the tx height:100
enter the distance between tx and rx :1000
enter the frequency:925000000
```



Inference

The path loss is decreases when height of receiver antenna is increased due to the factor [ahre]. This is because a higher antenna can potentially receive a stronger signal, reducing the overall path loss.

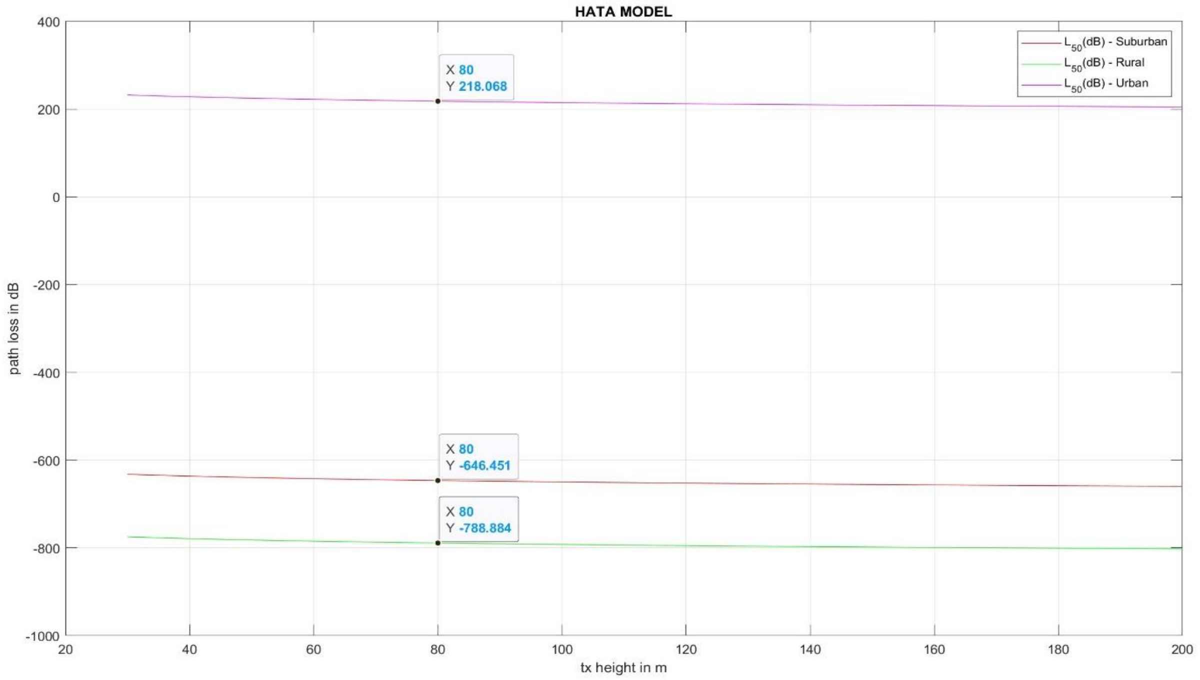
HATA Model transmitter height varied

Code

```
%HATA MODEL transmitter height varied
clc
clear all;
close all;
hre = input ('enter the rx height:');
hte = 30:200;
d=input ('enter the distance between tx and rx :');
f= input('enter the frequency:');

% MEDIUM-SMALL CITY (SUB URBAN AREA)

ahre_1=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_1=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_1;
L50_dB_1=L50_1-(2*log10(f/28)*log10(f/28))-5.4;

% MEDIUM SMALL CITY RURAL / OPEN AREA
ahre_2=(1.1*log10(f)-0.7)*hre-(1.56*log10(f)-0.8);
L50_2=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_2;
L50_dB_2=L50_2-(4.78*log10(f)*log10(f))+18.33*log10(f)-40.98;

% LARGE CITY URBAN AND SUB-URBAN AREA
if f <3000000000
   ahre_3=8.29*(log10(1.54*hre)).*(log10(1.54*hre))-1.1;
else
   ahre_3=3.2*(log10(11.75*hre)).*(log10(11.75*hre))-4.97;
end
L50_3=69.55+26.16*log10(f)-13.82*log10(hte)+(44.9-6.55*log10(hte))*log10(d)-ahre_3;
L50_dB_3=L50_3-(2*log10(f/28)*log10(f/28))-5.4;

% rx height vs path loss
figure
plot (hte, L50_dB_1,'r', hte, L50_dB_2,'g',hte, L50_dB_3,'m');
legend ('L_5_0(dB) - Suburban', 'L_5_0(dB) - Rural', 'L_5_0(dB) - Urban');
xlabel ('tx height in m');
ylabel ('path loss in dB');
title ('HATA MODEL');
grid on;
```

Output

```
enter the rx height:100
enter the distance between tx and rx :1000
enter the frequency:925000000
```



Inference

As the transmitter antenna height increases, the path loss generally decreases. This is because a higher transmitter antenna can potentially cover a larger area and provide a stronger signal to the receiver.

L_{50} (dB) of Urban City > L_{50} (dB) of Suburban City > L_{50} (dB) of Rural

Because of the reason that the obstacles and height of buildings are more in Urban city than Suburban and Rural places.

Experiment 5

Bit error rate analysis of digital communication receivers with Maximal Ratio Combining (MRC) receives diversity in frequency-flat and slowly varying fading channel.

Aim

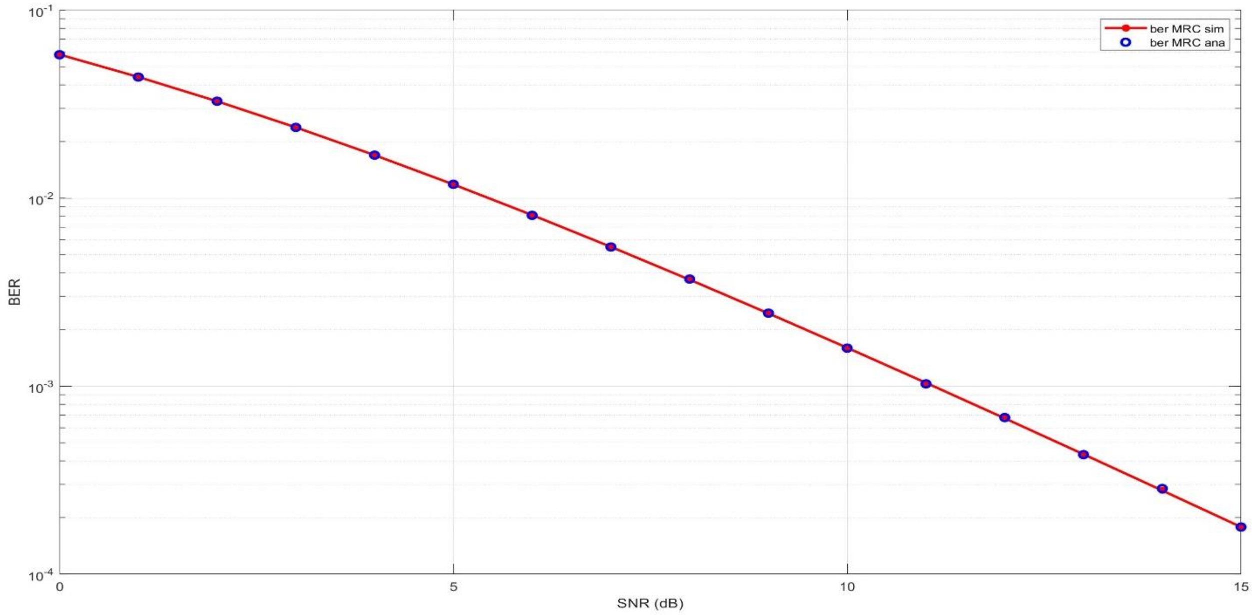
Bit error rate analysis of digital communication receivers with Maximal Ratio Combining (MRC) receive diversity in frequency-flat and slowly varying fading channel.

Code

```
clc;
clear all;
close all;
%%%%%%%%%%%%%%% Initialization %%%%%%%%%%%%%%%%%%%%%%%%%%%
N=5; % Number of trials
m = 10^6; %Number of bits in each trial
ip = rand(1,m)>0.5; % Generated bits
BPSK = 2*ip-1; % Generated BPSK symbols
snr_dB = 0:1:15; % range of snr values
snr = 10.^(snr_dB/10); % snr value in the normal scale
L=2; % Number of diversity branches

% theoretical BER value for MRC combiner with 2 diversity branches
p_R_MRC = 1/2 - 1/2*(1+1./snr).^(-1/2);
ber_MRC_ana = p_R_MRC.^2.*(1+2*(1-p_R_MRC));

%%%%%%%%%%% Receive MRC one by Two System %%%%%%%%%%%%%%%%%%%%%
n_err=zeros(1,length(snr_dB)); % Initialize the bit error counter
for p = 1:N
   for q = 1:length(snr_dB)
      % Generate white noise samples
      No = 1/sqrt(2)*[randn(L,m) + 1j*randn(L,m)];
      % Generate channel coefficient
      h = 1/sqrt(2)*[randn(L,m) + 1j*randn(L,m)];
      symbol = kron(ones(L,1),BPSK); % array of symbols
      rec_vector = h.*symbol + 10^(-snr_dB(q)/20)*No;% received symbol

      % Decision metric
      dec_metric = sum(conj(h).*rec_vector,1)./sum(h.*conj(h),1);
      ip_hat = real(dec_metric)>0; % Estimated symbol
      n_err(q) = n_err(q)+size(find([ip- ip_hat]),2); % compare input and symbols
   end
end

ber_MRC_sim = n_err/(N*m);
semilogy(snr_dB,ber_MRC_ana,'-r*','LineWidth',2)
hold on;
```

```
semilogy(snr_dB,ber_MRC_sim,'ob','LineWidth',2)
legend('ber MRC sim', 'ber MRC ana');
xlabel('SNR (dB)');
ylabel('BER');
grid on;
```

Output




```
ber_MRC_ana =

  Columns 1 through 9

    0.0581    0.0441    0.0328    0.0238    0.0169    0.0118    0.0081    0.0055    0.0037

  Columns 10 through 16

    0.0024    0.0016    0.0010    0.0007    0.0004    0.0003    0.0002


ber_MRC_sim =

  Columns 1 through 9

    0.0579    0.0441    0.0328    0.0238    0.0169    0.0118    0.0081    0.0055    0.0037

  Columns 10 through 16

    0.0024    0.0016    0.0010    0.0007    0.0004    0.0003    0.0002
```

Inference

Maximal Gain Combining is very efficient as the difference in the analytical value and simulated values are very less to zero.

Experiment 6

Bit error rate analysis of digital communication receivers with Equal Gain Combining (EGC) receives diversity in frequency-flat and slowly varying fading channel.

Aim

Bit error rate analysis of digital communication receivers with Equal Gain Combining (EGC) receive diversity in frequency-flat and slowly varying fading channel.

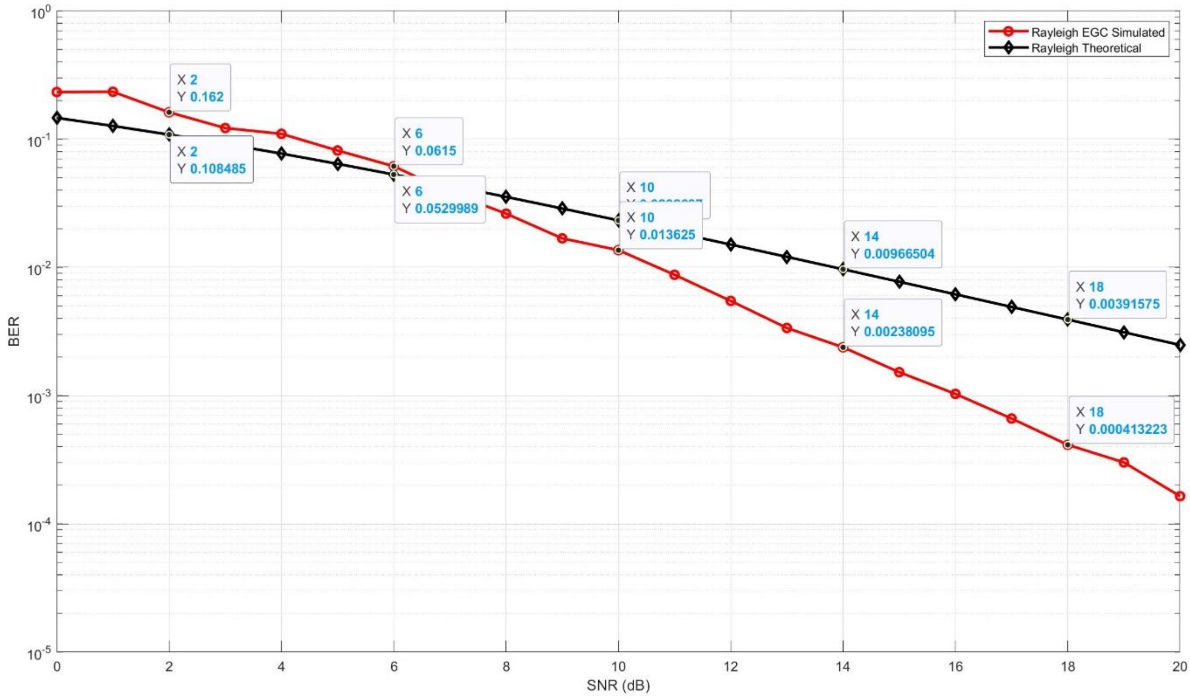
Code

```
clc;
close all;
clear all;
% Number of information bits
m= 10^3;
%Range of SNR values
snr_dB = [0:1:20];
snr = 10.^(snr_dB/10);
for j=1:1:length(snr_dB)
   n_err = 0;
   n_bits = 0;
   while n_err < 100
      inf_bits=round(rand(1,m));
      % BPSK modulator
      x=-2*(inf_bits-0.5);
      % Noise variance
      N0=1/10^(snr_dB(j)/10);
      n1 = sqrt(N0/2)*abs((randn(1,length(x))+ 1i*randn(1,length(x)))); %noise for the first
      n2 = sqrt(N0/2)*abs((randn(1,length(x))+ 1i*randn(1,length(x)))); %noise for the first
      h1 = sqrt(0.5)*abs((randn(1,length(x))+ 1i*randn(1,length(x)))); %rayleigh amplitude 1
      h2 = sqrt(0.5)*abs((randn(1,length(x))+ 1i*randn(1,length(x)))); %rayleigh amplitude 1
      %Equal Gain combining
      y1 = h1.*x+n1; % Signal 1
      y2 = h2.*x+n2; % Signal 2
      y_equal = 0.5*(y1+y2);
      % Decision making at the Receiver
      est_bits=y_equal < 0;
      % Calculate Bit Errors
      diff=inf_bits-est_bits;
      n_err=n_err+sum(abs(diff));
      n_bits=n_bits+length(inf_bits);
   end
   % Calculate Bit Error Rate
   BER(j)=n_err/n_bits;
end
```

```
theoryBer=0.5.*(1-sqrt(snr./(snr+1))); % Rayleigh Theoretical BER
semilogy(snr_dB,BER,'or-','LineWidth',2);
hold on;
semilogy(snr_dB,theoryBer,'blad-','LineWidth',2);
hold on;
legend('Rayleigh EGC Simulated', 'Rayleigh Theoretical');
axis([0 20 10^-5 1]);
xlabel('SNR (dB)');
ylabel('BER');
grid on;
```

Output

```
BER =

  Columns 1 through 9

    0.2330    0.2340    0.1620    0.1220    0.1100    0.0815    0.0615    0.0383    0.0262

theoryBer =

  Columns 1 through 9

    0.1464    0.1267    0.1085    0.0919    0.0771    0.0642    0.0530    0.0435    0.0355
```



Inference

As the SNR increases, the received signal power becomes stronger relative to the noise. The simulation includes practical factors like finite block length and quantization effects, which can lead to a higher BER compared to the theoretical BER. The EGC technique helps to mitigate the effects of fading by combining the signals from multiple antennas, leading to a lower BER compared to a single-antenna system.