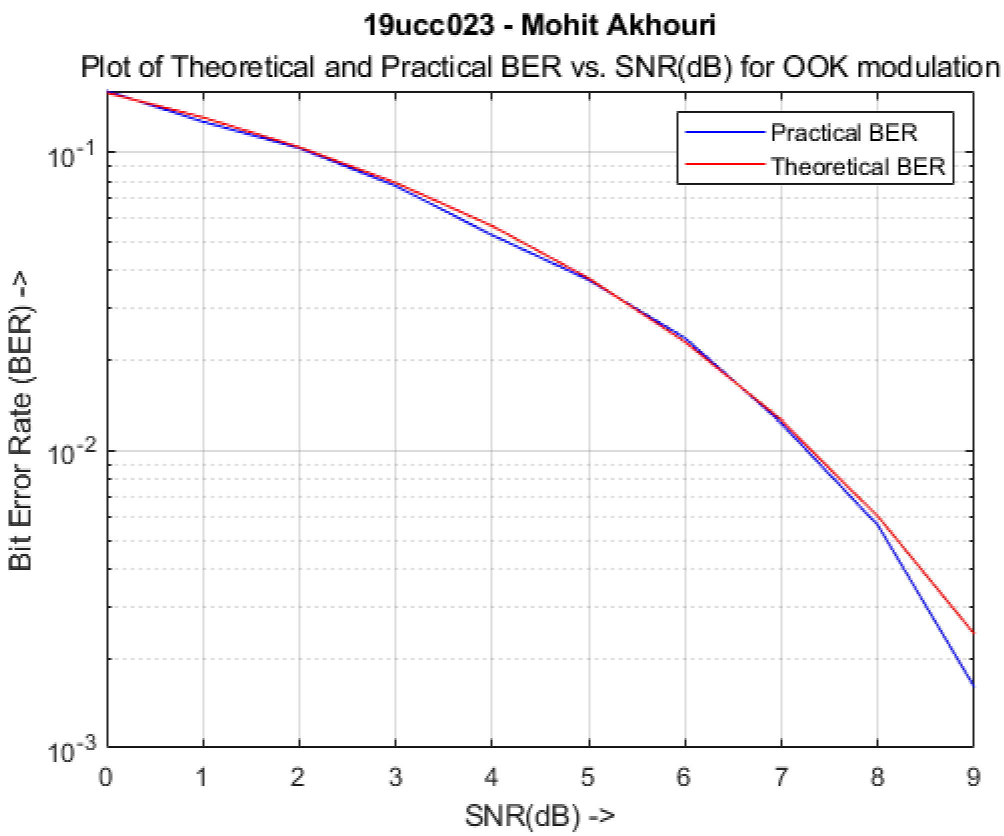
```
% 19ucc023
% Mohit Akhouri
% Observation 2 - Practical and Theoretical BER of OOK Modulation

% This code will implement OOK Modulation and compare the theoretical
 and
% analytical BER

% This code will also plot the graph between Theoretical and Practical
 BER
% vs. Signal to Noise Ratio ( SNR )

clc;
clear all;
close all;

size = 10000; % intializing the size for the random variable and input
 signal
BER_Practical = zeros(1,10); % Initializing the Array to store
 practical values of BER
BER_Theoretical = zeros(1,10); % Initializing the Array to store
 Theoretical values of BER

x=zeros(1,size); % Initializing the array to store the POLAR input
 signal x[n]

% ALGORITHM for initializing a UNI-POLAR SIGNALLING x[n]
for i=1:size
    rnd = rand();
    if(rnd>0.5)
        x(i)=1;  % +V in UNI-POLAR SIGNALLING
    else
        x(i)=0; % 0 in UNI-POLAR SIGNALLING
    end
end

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
for i=1:length(SNR_dB)

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

    y=zeros(1,size); % to store the output signal y[n] = x[n] + n , n=
 AWGN noise
    n=zeros(1,size); % to store the AWGN noise

    % Loop for calculation of AWGN noise and storing in variable 'n'
```

```
    for j=1:size
        n(j)=sqrt(1/2)*M*randn(); % using randn function to randomly
 choose any integer
    end

    % Loop to calculate the output signal y[n] = x[n] + n, n = AWGN
 noise
    for j=1:size
        y(j)=x(j)+n(j);
    end

    % Main Loop algorithm for ML-Detection of OOK modulation
    yn=zeros(1,size);
    for j=1:size
        if(y(j)>=0.5) % Based on decision rule , either +V(1) or 0 is
 choosen
            yn(j)=1;
        else
            yn(j)=0;
        end
    end

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

    BER_Practical(i)=BER_Practical(i)/size; % Calculation of Practical
 BER
    BER_Theoretical(i)=qfunc(sqrt(1/N));  % Calculation of Theoretical
 BER using Q function

end

% Display of Theoretical and Practical BER
disp(sprintf('%-10s \t %-20s \t %-20s','index','Theoretical
 BER','Practical BER'));
for i=1:10
    disp(sprintf('%-10i %-20d \t
 %-20d',i,BER_Practical(i),BER_Theoretical(i)));
end

% Plots of Practical and Theoretical BER vs. Signal to Noise Ratio
 ( SNR )
% in dB
semilogy(SNR_dB,BER_Practical,'Color','blue'); % semilogy used for
 plotting on base-10 logarithmic scale on Y-axis
hold on;
semilogy(SNR_dB,BER_Theoretical,'Color','red'); % semilogy used for
 plotting on base-10 logarithmic scale on Y-axis
```

```
ylabel('Bit Error Rate (BER) ->');
xlabel('SNR(dB) ->');
legend('Practical BER','Theoretical BER');
title('19ucc023 - Mohit Akhouri','Plot of Theoretical and Practical
 BER vs. SNR(dB) for OOK modulation');
grid on;

hold off;
```

```
index         Theoretical BER        Practical BER
1           1.607000e-01           1.586553e-01
2           1.267000e-01           1.309273e-01
3           1.032000e-01           1.040286e-01
4           7.700000e-02           7.889587e-02
5           5.260000e-02           5.649530e-02
6           3.720000e-02           3.767899e-02
7           2.370000e-02           2.300714e-02
8           1.230000e-02           1.258703e-02
9           5.600000e-03           6.004386e-03
10          1.600000e-03           2.413310e-03
```



Published with MATLAB® R2020b