
Wireless Comms mini Matlab 2

```
%Neema Aggarwal
%Shivam Mevawala
%Nicolas Castro

close all;
SNR = -4:2:20; %list of SNR values to run algorithm
%intialize vecs
BER_flat=zeros(length(SNR));
BER_sel=zeros(length(SNR));

n=3072; %number of samples
m=4; %QPSK is 4-QAM

delayVector = [0 1 2 3 4] * 1e-5; % Discrete delays of four-path channel (s)
gainVector = [0 -4 -6 -9 -14];

rchan_flat=rayleighchan(1e-5,1e4); %Set up the channel fading object
rchan_flat.StoreHistory = 1; %If this value is 1, channel state information
%needed by the channel visualization tool is stored.
rchan_flat.StorePathGains = 1;%If set to 1, the complex path gain vector is
%stored.

rchan_sel=rayleighchan(1e-5,1, delayVector, gainVector);%Set up the
%channel fading object with delay and gain vecs
rchan_sel.StoreHistory = 1;
rchan_sel.StorePathGains = 1;

%use the SNR to calculate EbNo for the normal sytem and the convolutional
%coder
EbNo = SNR -10*log10(log2(m));
eq = dfe(5, 5, rls(.99)); %Construct a decision feedback equalizer object
eq.SigConst=qammod(0:3,4); %Set the constellation to 4-qam
bers=zeros(1,10); %initialize bers

%loop over SNR values
for k=1:length(SNR)
    %generate a random vector of 4 symbols
    X=randi([0 m-1],1,n);
    %modulate
    Y=qammod(X,m);
    %add noise and rayleigh fading
    A=filter(rchan_flat,Y);
    A = awgn(A, SNR(k), 'measured');
    Ae=A./rchan_flat.PathGains.';
    %demodulate
    Z=qamdemod(Ae,m);
    %calculate bit error rate
    BER_flat(k)=biterr(Z,X)/(2*n);

    %Frequency selective channel mitigation
```

```

    for kk=1:50
        %add noise and rayleigh fading
        As=filter(rchan_sel,Y);
        As = awgn(As, SNR(k), 'measured');
        %equalize
        Ase=equalize(eq,As,Y(1:300));
        %demod
        Zs=qamdemod(Ase,m);
        %calculate the BER
        bers(kk)=biterr(Zs,X)/(2*n);
    end
    BER_sel(k)=mean(bers);
end

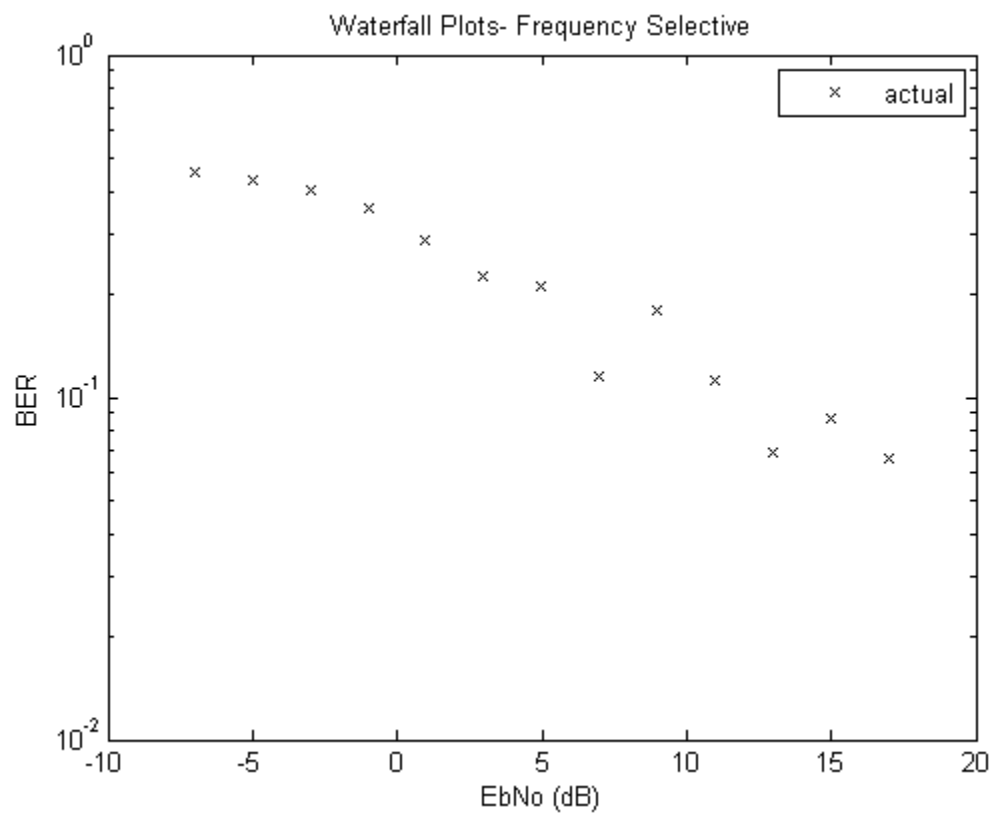
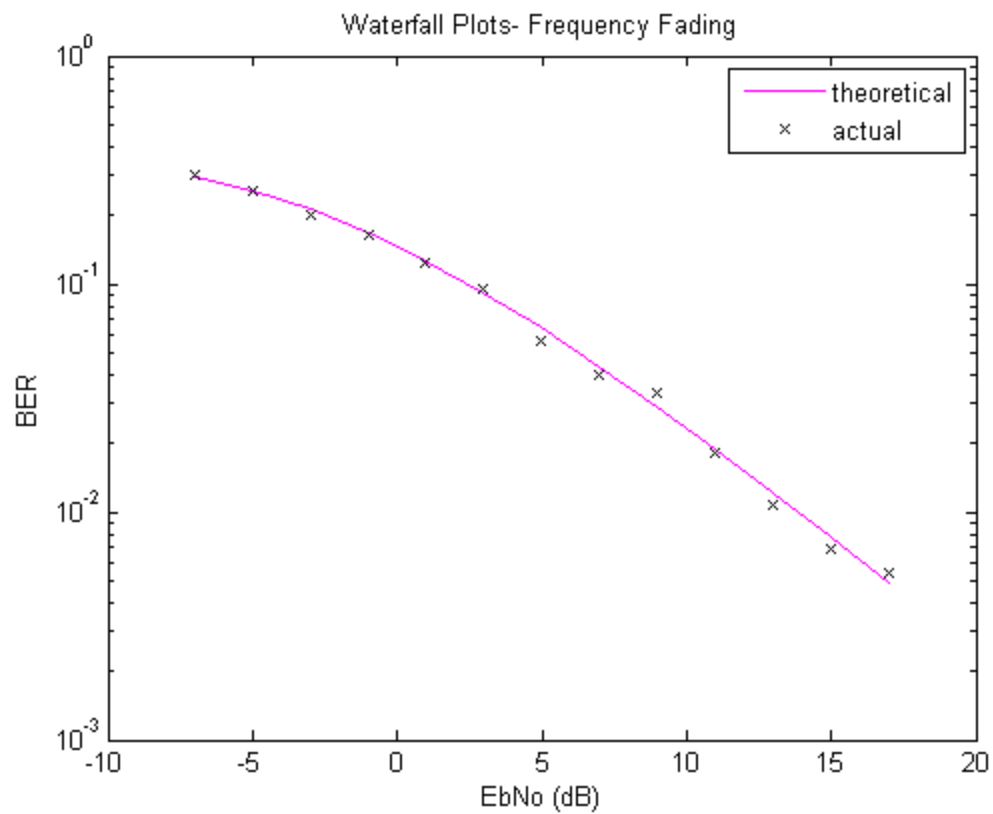
%plots
figure
semilogy(EbNo,berfading(EbNo, 'qam',4,1), 'm-');
hold on;
semilogy(EbNo, BER_flat, 'kx');

xlabel('EbNo (dB)')
ylabel('BER')
title('Waterfall Plots- Frequency Fading')
legend('theoretical', 'actual')

figure
semilogy(EbNo, BER_sel, 'kx');

xlabel('EbNo (dB)')
ylabel('BER')
title('Waterfall Plots- Frequency Selective')
legend('actual')

```



Published with MATLAB® R2013b