





































clear all; clc;

n=10; %Number of data symbols

Tsym=8; %Symbol time interms of sample time or oversampling rate equivalently

SNRstep=5;

SNR\_dB=-20:SNRstep:20;

rng('default');%set the random generator seed to default.

BinData=round(rand(1,n));

data=2\*BinData-1; %BPSK data Unipolar 2 Bipolar

bpsk=reshape(repmat(data,1,Tsym)',n\*Tsym,1); %BPSK signal

L=length(bpsk);

PDc1=0;

PDc2=0;

PFc1=0;

PFc2=0;

PFc=0;

PFs=0.1;

for i=1:length(SNR\_dB)

SNR = 10^(SNR\_dB(i)/10); %SNR to linear scale

Esym=sum(abs(bpsk).^2)/(L); %Calculate actual symbol energy

N0=Esym/SNR; %Find the noise spectral density

noiseSigma = sqrt(N0);%Standard deviation for AWGN Noise when x is real

noise(:,i) = noiseSigma\*randn(1,L);%computed noise

receivedx(:,i)=bpsk+noise(:,i) ;

receivedfliped=flipud(receivedx(:,i));

impRes = [0.5 ones(1,6) 0.5]; %Averaging Filter -&gt; u[n]-u[n-Tsamp]

yy1(:,i)=conv(receivedx(:,i),impRes,'full');

xp(:,i)=receivedfliped;

thresh(:,i)=noise(:,i).\*xp(:,i);

yy2(:,i)=filter(xp(:,i),1,receivedx(:,i));

for j = 1:L

if(yy1(j,i)>thresh(j,i))

PDc1=PDc1+1;

end

if(yy2(j,i)>thresh(j,i))

PDc2=PDc2+1;

end

if(noise(j,i)>thresh(j,i))

PFc=PFc+1;

end

if(yy1(j,i)<thresh(j,i))

PFc1=PFc1+1;

end

if(yy2(j,i)<thresh(j,i))

PFc2=PFc2+1;

end

end

PD1(:,i)=PDc1/(L\*n);

PD2(:,i)=PDc2/(L\*n);

PF1(:,i)=PFc1/(L\*n);

PF2(:,i)=PFc2/(L\*n);

PFinv(:,i)=(PFc/(L\*n));

PF(:,i)=1-PFinv(:,i);

TheoryThreshold(i)=qfuncinv(PFinv(:,i)).\*sqrt(SNR\*noiseSigma^2);

PFstatic(i)=PFs;

StaticThreshold(i)=qfuncinv(PFs).\*sqrt(SNR\*noiseSigma^2);

PDtheory(i)=qfunc((TheoryThreshold(i)-SNR)/sqrt(SNR\*noiseSigma^2));

PMdy(i)=1-PDtheory(i);

PDstatictheory(i)=qfunc((StaticThreshold(i)-SNR)/sqrt(SNR\*noiseSigma^2));

PMstatic(i)=1-PDstatictheory(i);

PFAtheory(i)=qfunc((1-TheoryThreshold(i))/sqrt(SNR\*noiseSigma^2));

figure('Color',[1 1 1]);

subplot(4,1,1);

plot(bpsk);

titleX=sprintf('Simulation with %d dB',SNR\_dB(i));

title(titleX);

xlabel('Sample index [n]');

ylabel('Amplitude');

set(gca,'XTick',0:8:L);

axis([1 L -2 2]); grid on;

subplot(4,1,2);

plot(receivedx(:,i));

title('Transmitted BPSK symbols (with noise)');

xlabel('Sample index [n]');

ylabel('Amplitude')

ymax=max(receivedx(:,i))+1;

ymin=min(receivedx(:,i))+1;

set(gca,'XTick',0:8:L);

axis([1 L ymin ymax]); grid on;

subplot(4,1,3);

plot(yy1(:,i));

title('Matched Filter (Averaging Filter) output');

xlabel('Sample index [n]');

ylabel('Amplitude');

set(gca,'XTick',0:8:L);

ymax=max(yy1(:,i))+1;

ymin=min(yy1(:,i))+1;

axis([1 L ymin ymax]); grid on;

subplot(4,1,4);

plot(yy2(:,i));

title('Matched Filter (Rational IR) output');

xlabel('Sample index [n]');

ylabel('Amplitude');

set(gca,'XTick',0:8:L);

ymax=max(yy2(:,i))+1;

ymin=min(yy2(:,i))+1;

axis([1 L ymin ymax]); grid on;

pause(1);

end

figure

plot(SNR\_dB,PD1,'-ro',...

'LineWidth',2,'MarkerEdgeColor','k',...

'MarkerFaceColor','g','MarkerSize',5);

hold on

plot(SNR\_dB,PD2,'-bo',...

'LineWidth',2,'MarkerEdgeColor','y',...

'MarkerFaceColor','g','MarkerSize',5);

legend('Averaging Filter','Rational IR Filter')

ylabel('Probability of Detection P\_D');

xlabel('SNR\_d\_B');

title('Probability of Detection');

grid on;

figure

plot(SNR\_dB,PDtheory,'-go',...

'LineWidth',2,'MarkerEdgeColor','k',...

'MarkerFaceColor','g','MarkerSize',5);

ylabel('Probability of Detection P\_D');

xlabel('SNR\_d\_B');

title('Theoritical P\_D');

grid on;

figure

plot(SNR\_dB,PF,'-mo',...

'LineWidth',2,'MarkerEdgeColor','g',...

'MarkerFaceColor','g','MarkerSize',5);

ylabel('Probability of False Alarm P\_F\_A');

xlabel('SNR\_d\_B');

title('Probability of False Alarm');

grid on;

PM1=1-PD1;

PM2=1-PD2;

figure

plot(SNR\_dB,PM1,'-ro',...

'LineWidth',2,'MarkerEdgeColor','k',...

'MarkerFaceColor','g','MarkerSize',5);

hold on

TheoryThreshold=TheoryThreshold(end:-1:1);

plot(SNR\_dB,PM2,'-bo',...

'LineWidth',2,'MarkerEdgeColor','y',...

'MarkerFaceColor','g','MarkerSize',5);

legend('Averaging Filter','Rational IR Filter')

ylabel('Probability of Miss Ditection P\_M');

xlabel('SNR\_d\_B');

title('Probability of Miss Ditection');

grid on;

TER1=PM1+PF;

TER2=PM2+PF;

figure

plot(SNR\_dB,TER1,'-bo',...

'LineWidth',2,'MarkerEdgeColor','k',...

'MarkerFaceColor','g','MarkerSize',5);

hold on

plot(SNR\_dB,TER2,'-go',...

'LineWidth',2,'MarkerEdgeColor','y',...

'MarkerFaceColor','g','MarkerSize',5);

legend('Averaging Filter','Rational IR Filter')

ylabel('TER');

xlabel('SNR\_d\_B');

title('TOtal Error rate');

grid on;

figure

plot(SNR\_dB,PFAtheory,'-yo',...

'LineWidth',2,'MarkerEdgeColor','r',...

'MarkerFaceColor','g','MarkerSize',5);

ylabel('Probability of False Alarm P\_F\_A');

xlabel('SNR\_d\_B');

title('Probability of False Alarm Theoritical');

grid on;

figure

plot(thresh(:,1),'LineWidth',1.5)

hold on

plot(thresh(:,2),'LineWidth',1.5)

plot(thresh(:,3),'LineWidth',1.5)

plot(thresh(:,4),'LineWidth',1.5)

plot(thresh(:,5),'LineWidth',1.5)

plot(thresh(:,6),'LineWidth',1.5)

plot(thresh(:,7),'LineWidth',1.5)

plot(thresh(:,8),'LineWidth',1.5)

plot(thresh(:,9),'LineWidth',1.5)

legend('-20 dB','-15 dB','-10 dB','-5 dB','0 dB','5 dB','10 dB','15 dB','20 dB')

xlabel('Time Index')

ylabel('Energy')

title('Simulated Dynamic Thresholds')

figure

plot(SNR\_dB,StaticThreshold,'-m\*',...

'LineWidth',2,'MarkerEdgeColor','b',...

'MarkerFaceColor','g','MarkerSize',5);

hold on

plot(SNR\_dB,TheoryThreshold,'-yo',...

'LineWidth',2,'MarkerEdgeColor','k',...

'MarkerFaceColor','g','MarkerSize',5);

legend('Static Threshold','Dynamic Threshold')

ylabel('THreshold');

xlabel('SNR\_d\_B');

title('THreshold');

grid on;

figure

plot(SNR\_dB,PDstatictheory,'LineWidth',1.5)

hold on

plot(SNR\_dB,PDtheory,'LineWidth',1.5)

legend('Static','Dynamic')

ylabel('P\_D');

xlabel('SNR\_d\_B');

title('P\_D Static vs Dynamic')

figure

plot(SNR\_dB,PFstatic,'LineWidth',1.5)

hold on

plot(SNR\_dB,PF,'LineWidth',1.5)

legend('Static','Dynamic')

ylabel('P\_F\_A');

xlabel('SNR\_d\_B');

title('P\_F\_A Static vs Dynamic')

figure

plot(SNR\_dB,PMstatic,'LineWidth',1.5)

hold on

plot(SNR\_dB,PMdy,'LineWidth',1.5)

legend('Static','Dynamic')

ylabel('P\_M');

xlabel('SNR\_d\_B');

title('P\_M Static vs Dynamic')

