**STBC**

close all;

clear all;

N=input('enter the length of input bit stream');

x=randi([0 1],1,N);

t=pskmod(x,2);

h=1/sqrt(2)\*(randn(1,N)+i\*randn(1,N));

n=1/sqrt(2)\*[randn(1,N) + j\*randn(1,N)];

for snr=1:30

sCode = zeros(2,N);

sCode(:,1:2:end) = (1/sqrt(2))\*reshape(t,2,N/2); % [x1 x2 ...]

sCode(:,2:2:end) = (1/sqrt(2))\*(kron(ones(1,N/2),[-1;1]).\*flipud(reshape(conj(t),2,N/2)));

hMod = kron(reshape(h,2,N/2),ones(1,2));

y=sum(hMod.\*sCode,1)+10^(-snr/20)\*n;

hEq = zeros(2,N);

hEq(:,[1:2:end]) = reshape(h,2,N/2); % [h1 0 ... ; h2 0...]

hEq(:,[2:2:end]) = kron(ones(1,N/2),[1;-1]).\*flipud(reshape(h,2,N/2));

hEq(1,:) = conj(hEq(1,:)); % [h1\* h2\* ... ; h2 -h1 .... ]

hEqPower = sum(hEq.\*conj(hEq),1);

yMod = kron(reshape(y,2,N/2),ones(1,2));

yMod(2,:) = conj(yMod(2,:));

r = sum(hEq.\*yMod,1)./hEqPower;

r(2:2:end) = conj(r(2:2:end));

r1=pskdemod(r,2);

[num,ratio]=biterr(r1,x);

g(snr)=ratio;

if g(snr)==0

g(snr)=10e-7;

end

end

snr=1:30;

EbN0 = 10.^(snr/10);

theoryBer= 0.5.\*(1-1\*(1+1./EbN0).^(-0.5));

semilogy(snr,g);

hold on;

semilogy(snr,theoryBer);

axis([0 30 0.0000001 1])

legend('simulated ber','theoretical ber');

xlabel('snr');

ylabel('BER');

title('STBC');