* **Simulation study of various Entropies and mutual information in a communication system**

Code 1)

Po= 0:0.01:1;

H\_Po= zeros(1,length(Po));

for i= 2:length(Po)-1

H\_Po(i)= -Po(i)\*log2(Po(i))-(1-Po(i))\*log2(1-Po(i));

end

plot2d(Po,H\_Po)

xlabel('Symbol Probability, Po')

ylabel('H(Po)')

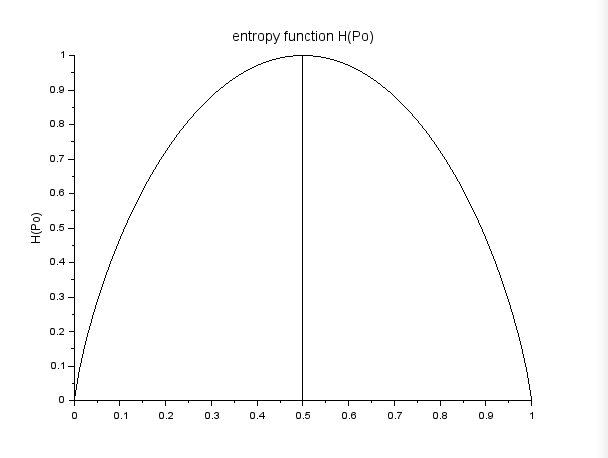
abort

ylabel('H(Po)')

title('entropy function H(Po)')

plot2d3('gnn',0.5,1)

Output:



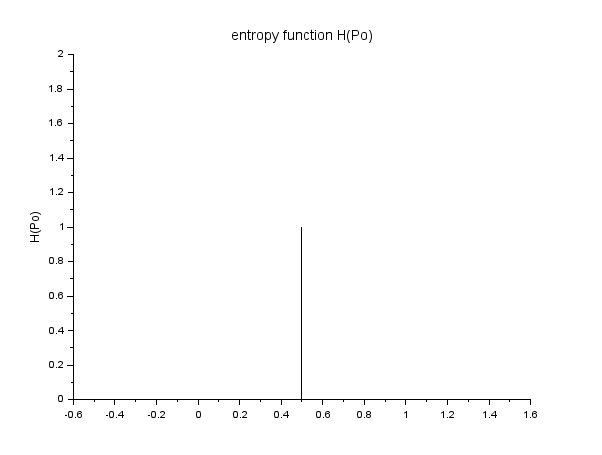
Code 2:

ylabel('H(Po)')

title('entropy function H(Po)')

plot2d3('gnn',0.5,1)

Output:



Code 3:

Po= 0:0.01:1;

H\_Po= zeros(1,length(Po));

for i= 2:length(Po)-1

H\_Po(i)= -Po(i)\*log2(Po(i))-(1-Po(i))\*log2(1-Po(i));

end

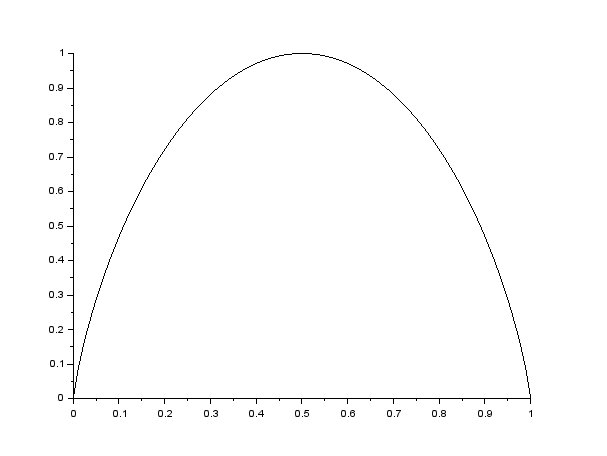
plot2d(Po,H\_Po)

xlabel('Symbol Probability, Po')

ylabel('H(Po)')

abort

Output:



* **Simulation of Performance of Digital Communication system with error control coding**

Code:

D = poly(0,'D');

g1D = 1+D+D^2;

g2D = 1+D^2;

mD = 1+0+0+D^3+D^4;

x1D = g1D\*mD;

x2D = g2D\*mD;

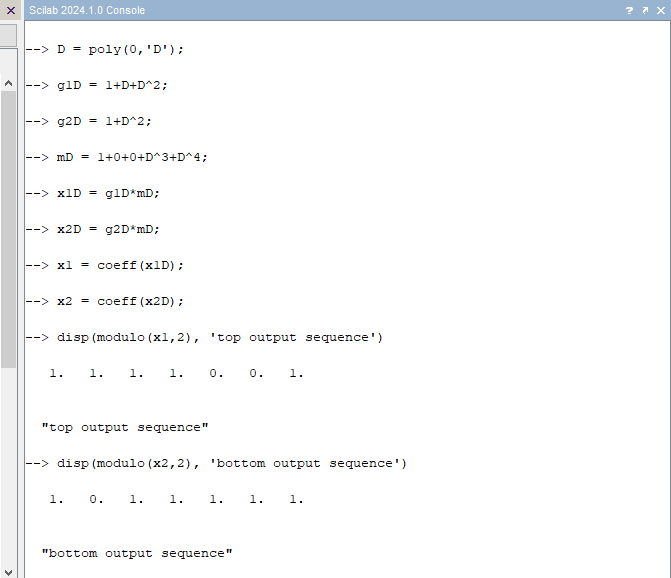
x1 = coeff(x1D);

x2 = coeff(x2D);

disp(modulo(x1,2), 'top output sequence')

disp(modulo(x2,2), 'bottom output sequence')

Output:



* **Simulation of Convolutional Coding**

Code:

r=1/2;

n=2;

pe=0.04;

p=1-pe;

gamma\_1 = 2\*log2(p)+2\*(1-r);

gamma\_2 = log2(pe\*p)+1;

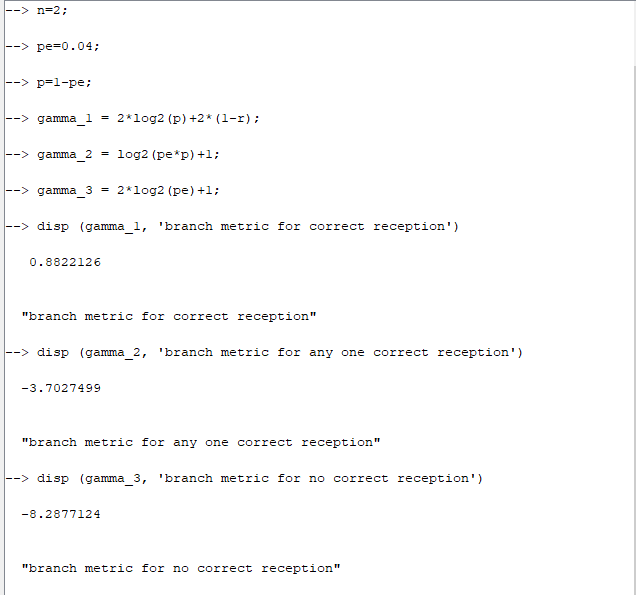
gamma\_3 = 2\*log2(pe)+1;

disp (gamma\_1, 'branch metric for correct reception')

disp (gamma\_2, 'branch metric for any one correct reception')

disp (gamma\_3, 'branch metric for no correct reception')

Output:



* **Simulation study of performance BPSK receiver in presence of noise.**

Code:

clear;

clc;

P\_S = 1

P\_A = 0.2

P\_B = 0.4

P\_AUB = 0.5

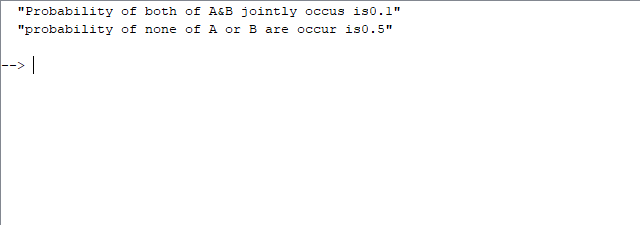
P\_AinterB = P\_A+P\_B-P\_AUB

disp ('Probability of both of A&B jointly occus is' +string(P\_AinterB))

P\_NOAB = P\_S-P\_AUB

disp('probability of none of A or B are occur is' +string(P\_NOAB))

Output:



* **Simulation study of Random processes. Find various statistical parameters of random processes.**

Code:

function **X**=f(**x**),

z=3\*(1-**x**)^2,

**X**=**x**\*z

endfunction

a=0;

b=1;

EX=intg(a,b,f);

function **Y**=c(**y**),

z=3\*(1-**y**)^2,

**Y**=**y**\*z

endfunction

EY=intg(a,b,c);

disp(EX, "i)Mean of X =");

disp(EX, "Mean of Y =");

function **X**=g(**x**),

z=3\*(1-**x**)^2,

**X**=**x**^2\*z

endfunction

a=0;

b=1;

EX2=intg(a,b,g);

function **Y**=h(**y**),

z=3\*(1-**y**)^2,

**Y**=**y**^2\*z

endfunction

EY2= intg(a,b,h);

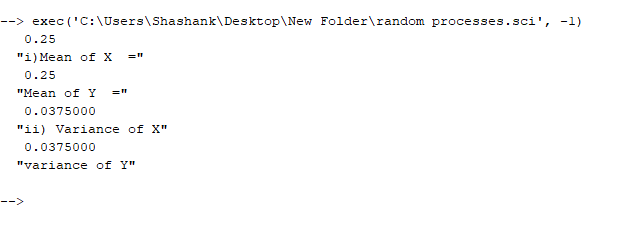
vX2=EX2-(EX)^2;

vY2=EY2-(EY)^2;

disp(vX2, "ii) Variance of X");

disp(vY2, "variance of Y");

Output:



* **Simulation study of Performance of M-ary QAM.**

Code: