

School of Engineering and Applied Science (SEAS)
Ahmedabad University

BTech(ICT) Semester V: Wireless Communication (ECE311)

Laboratory Assignment-6

Enrollment No: AU1841131

Name: Mansi Dobariya

1. Solution Problem-1

(a) Matlab Script:

```
1 close all;
2 clear all;
3 clc;
4 %bpsk
5 %step1: Analytical plot
6 %define SNR
7 Eb_No_db= [0:12] ;
8
9 %convert into linear
10 Eb_No_Lin=10.^(Eb_No_db/10);
11
12 %simulation
13 N=1000000;
14 ip = rand(1,N)>0.5; % generating 0,1 with equal probability
15 s = 2*ip-1; % BPSK modulation 0 -> -1; 1 -> 1
16 index=1;
17
18 for ii = 1:length(Eb_No_db)
19
20     p=sqrt(1/Eb_No_Lin(ii))
21     n1= 1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; % white gaussian noise, 0dB variance
22     h1=1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; %rayleigh channel
23     % Noise addition
24     y1 = h1.* s + p*n1 ; % additive white gaussian noise
25
26     n2= 1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; % white gaussian noise, 0dB variance
27     h2=1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; %rayleigh channel
28     % Noise addition
29     y2 = h2.* s + p*n2 ; % additive white gaussian noise
30
31     n3= 1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; % white gaussian noise, 0dB variance
32     h3=1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; %rayleigh channel
33     % Noise addition
34     y3 = h3.* s + p*n3 ; % additive white gaussian noise
35
36     n4= 1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; % white gaussian noise, 0dB variance
37     h4=1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; %rayleigh channel
38     % Noise addition
39     y4 = h4.* s + p*n4 ; % additive white gaussian noise
40
41     %mrc_1x3
42     for kk=1:N
43         d(kk)=conj(h1(kk)) * y1(kk) + conj(h2(kk)) * y2(kk)+ conj(h3(kk)) * y3(kk);
44         if (real(d(kk))>0)
45             dataDetect(kk)=1;
46         else
47             dataDetect(kk)=0;
48         end
49     end
50 end
```

```

51 %error detection
52 error=xor(ip,dataDetect);
53 bers(index)=sum(error)/N;
54 snr(index)=ii;
55 [snr(index) bers(index)]
56
57 %mrc_1x4
58 for kk=1:N
59 d2(kk)=conj(h1(kk)) * y1(kk) + conj (h2 (kk)) * y2(kk)+ conj (h3 (kk)) * y3(kk)
60 + conj (h4 (kk)) * y4(kk);
61 if (real(d2(kk))>0)
62 dataDetect2(kk)=1;
63 else
64 dataDetect2(kk)=0;
65 end
66 end
67
68 %error detection
69 error2=xor(ip,dataDetect2);
70 bers2(index)=sum(error2)/N;
71 snr(index)=ii;
72 [snr(index) bers2(index)]
73
74 %normalization
75 H1=h1.*conj(h1);
76 H2=h2.*conj(h2);
77 H3=h3.*conj(h3);
78 H4=h4.*conj(h4);
79
80 %antenna selection conditions
81 %sc_1x3
82 for mm=1:N
83 if (H1(mm)>H2(mm) && H1(mm)>H3(mm))
84 h(mm) = h1(mm);
85 y(mm) = y1(mm);
86 elseif (H2(mm)>H1(mm) && H2(mm)>H3(mm))
87 h(mm) = h2(mm);
88 y(mm) = y2(mm);
89 else
90 h(mm) = h3(mm);
91 y(mm) = y3(mm);
92 end
93 %error detection
94 d(mm) = conj(h(mm))*y(mm);
95 if(real(d(mm))>=0)
96 det(mm) = 1;
97 else
98 det(mm) = 0;
99 end
100 end
101 b1 = xor(ip, det);
102 sc_1x3(index) = sum(b1)/N;
103 snr(index)=ii;
104 [snr(index) sc_1x3(index)]
105
106 %antenna selection conditions
107 %sc_1x4
108 for mm=1:N
109 if (H1(mm)>H2(mm) && H1(mm)>H3(mm) && H1(mm)>H4(mm) )
110 h(mm) = h1(mm);
111 y(mm) = y1(mm);
112 elseif (H2(mm)>H1(mm) && H2(mm)>H3(mm) && H2(mm)>H4(mm))
113 h(mm) = h2(mm);
114 y(mm) = y2(mm);
115 elseif (H3(mm)>H1(mm) && H3(mm)>H2(mm) && H3(mm)>H4(mm))
116 h(mm) = h3(mm);
117 y(mm) = y3(mm);
118 else
119 h(mm) = h4(mm);
120 y(mm) = y4(mm);

```

```

120         end
121         %error detection
122         d4(mm) = conj(h(mm))*y(mm);
123         if(real(d4(mm))>=0)
124             det(mm) = 1;
125         else
126             det(mm) = 0;
127         end
128     end
129     b1_4 = xor(ip, det);
130     sc_1x4(index) = sum(b1_4)/N;
131     snr(index)=ii;
132     [snr(index) sc_1x4(index)]
133     index=index+1;
134 end
135
136 semilogy(Eb_No_db,bers,'d-','LineWidth',3);
137 hold on
138 semilogy(Eb_No_db,bers2,'d-','LineWidth',3);
139 hold on
140 semilogy(Eb_No_db,sc_1x3,'<-','LineWidth',3);
141 hold on
142 semilogy(Eb_No_db,sc_1x4,'<-','LineWidth',3);
143 grid on
144 legend('simulation MRC (1x3)', 'simulation MRC (1x4)', 'simulation SC (1x3)', '
simulation SC (1x4)');
145 xlabel('Eb/No, dB');
146 ylabel('Bit Error Rate');

```

(b) Simulation Output:

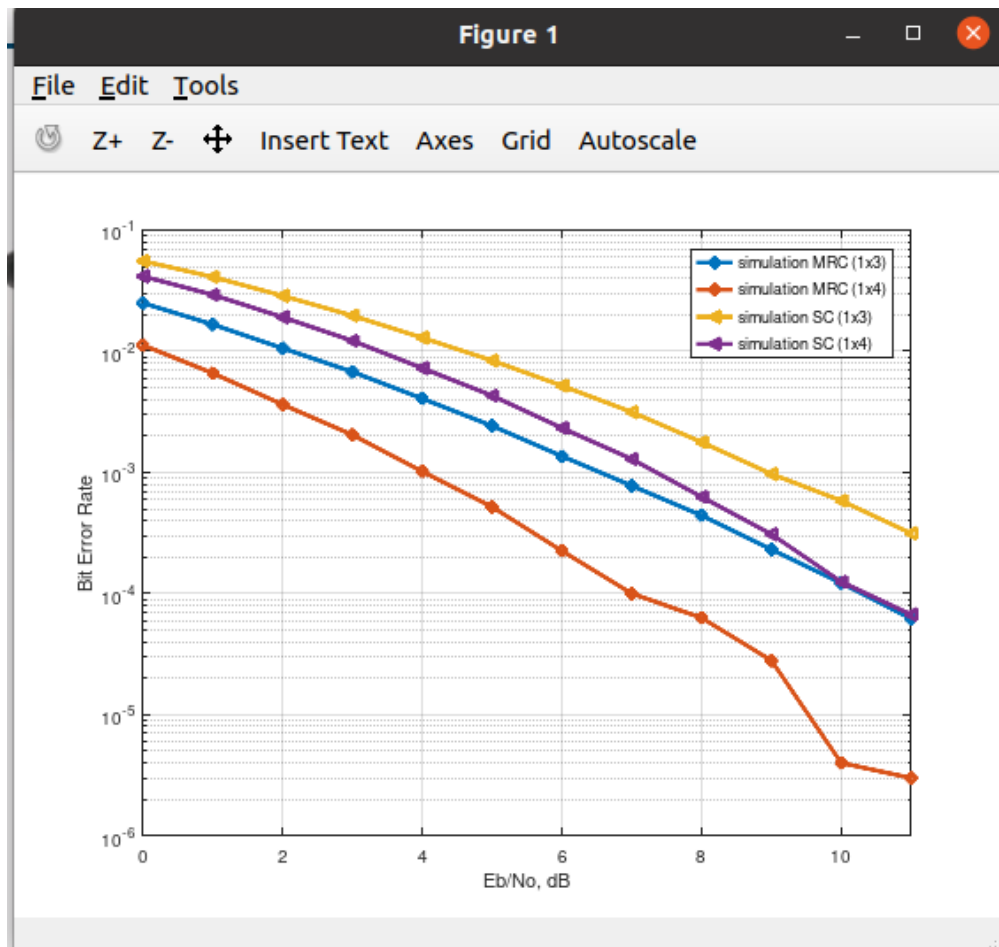


Fig 1:Plot of BER v/s SNR comparison with MRC and SC

(c) Inference:

- ★ MRC BER performs better than SC BER. As increasing no of antennas BER performs better with same SNR Value.
- ★ BER :: $MRC(1X4) > MRC(1X3) > SC(1X4) > SC(1X3)$
- ★ Reason is, MRC uses perfect knowledge of CSI , Selection Combining (SC) combines antennas at Rx as one switch