COMM 2 LAB 3 REPORT

UTKU ACAR / 250206062

3.1.c

I have made a mistake by creating signals. I wrote s0(length(tb)) and s1(length(tb)) instead of s0(1: length(tb)) and s1(1: length(tb)). Instead of starting 1 assigning values, I did assign just 100th value of signals as 1 or -1 with this way. This mistake effects all other results but when it's corrected all other values gets corrected easily. Corrected graphs have been given in right side of corresponding graphs.

3.3.c

SNR(P_signal/P_noise) affected the received signal so that values normally oscillate between -1 and 1 but with SNR these oscillation intervals become like below:

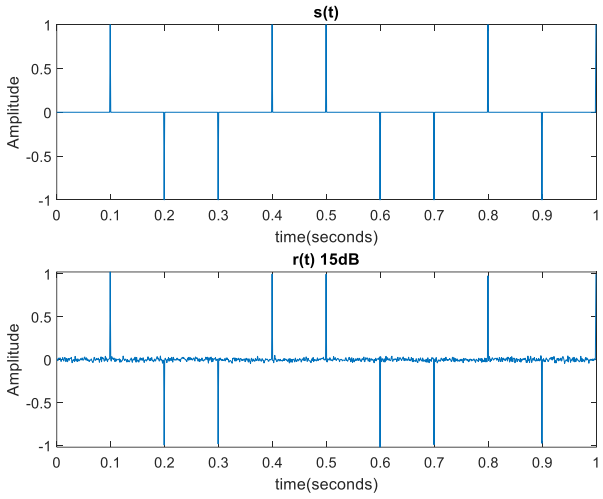
For 15 dB →[-1.4440,1.5346] →%53 error

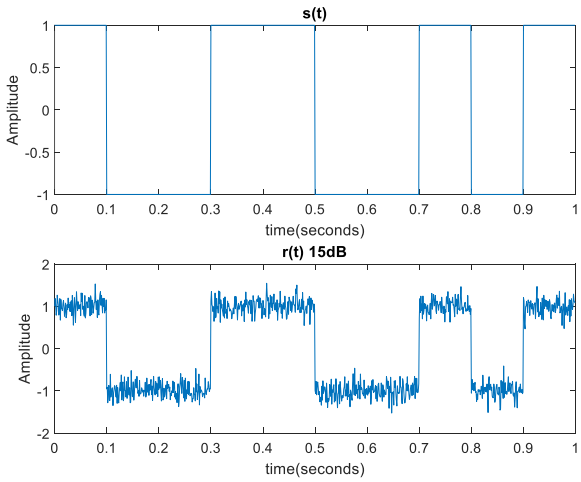
For 0 dB → [-3.6543, 3.8174] →%381 error

For -15dB→[-19.0784, 16.3485] →%1907 error

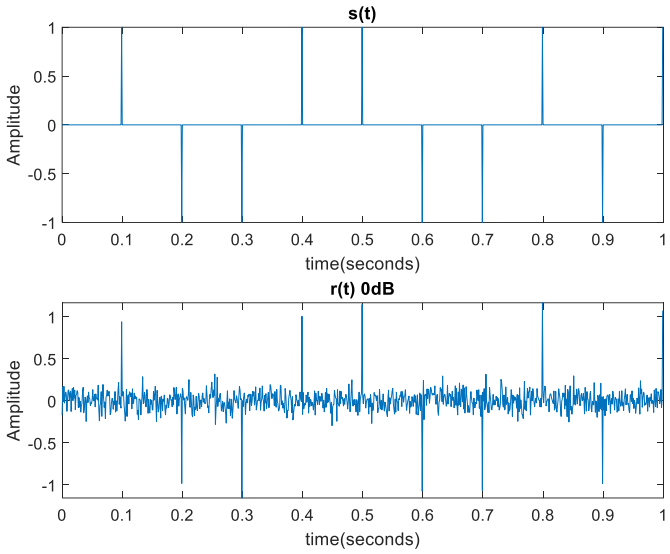
Which are unacceptable error rates. Graphical expressions have been given for (-15,0,15) dB SNRs below. (3.5.b.1, 3.5.b.2, 3.5.b.3)

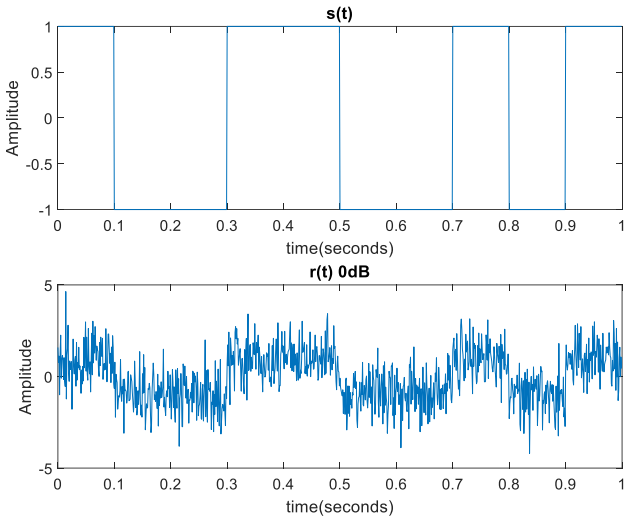
3.5.b.1



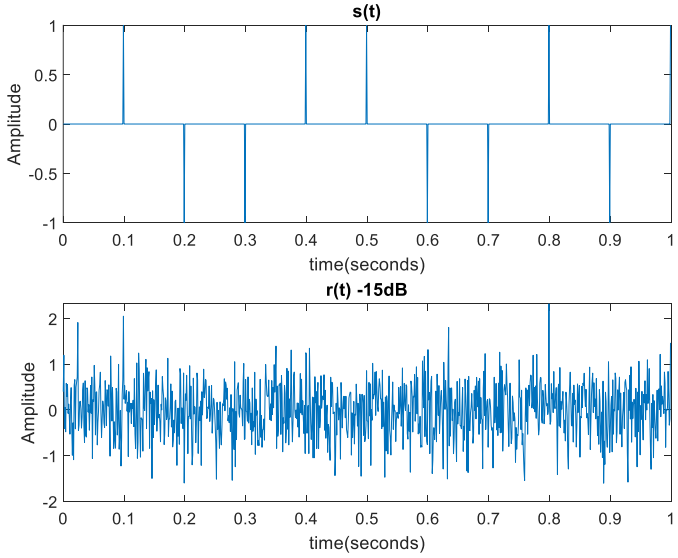


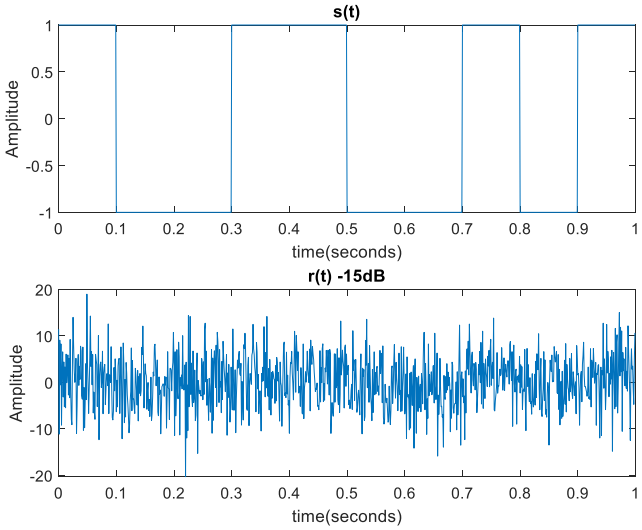
3.5.b.2


s(t)
Amplitude
time(seconds)
r(t) 0dB
Amplitude
time(seconds)


s(t)
Amplitude
time(seconds)
r(t) 0dB
Amplitude
time(seconds)

3.5.b.3


s(t)
Amplitude
time(seconds)
r(t) -15dB
Amplitude
time(seconds)


s(t)
Amplitude
time(seconds)
r(t) -15dB
Amplitude
time(seconds)

3.5.c

As we can see in plots 3.5.c.1-3 when SNR value decreases (Noisier signal), rate of the number of received Symbol decreases too This rate is stable while we are having SNR as 15 dB but while we are having 0 dB SNR rate has becoming lower this is getting more lower while we are having -15 dB SNR. Ratios given below:

Standard rate is 100

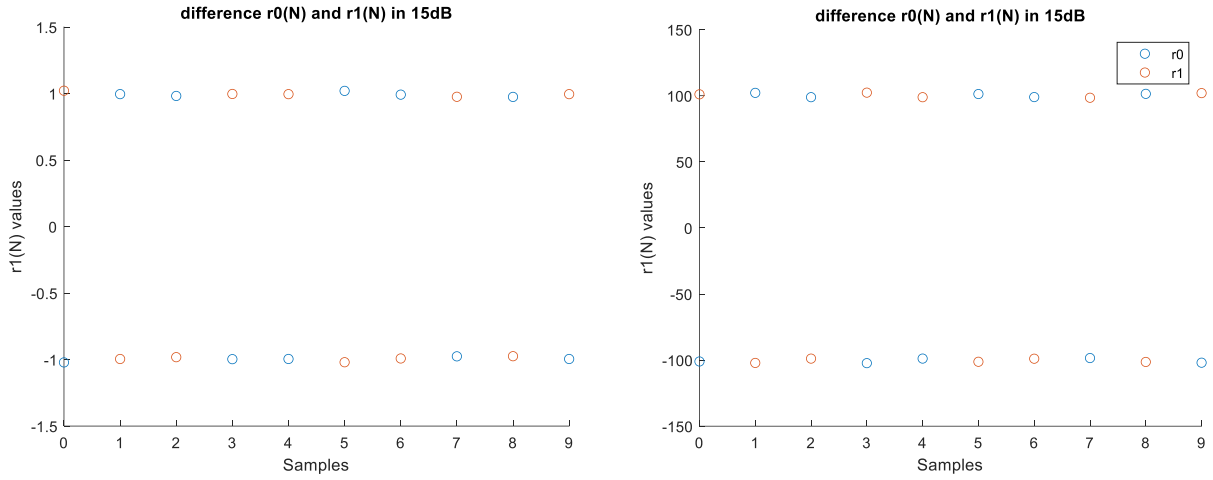
For 15 dB →[102.406,102.406] →%2.4 error

For 0 dB → [-110.44, 110.44] →%10.44 error

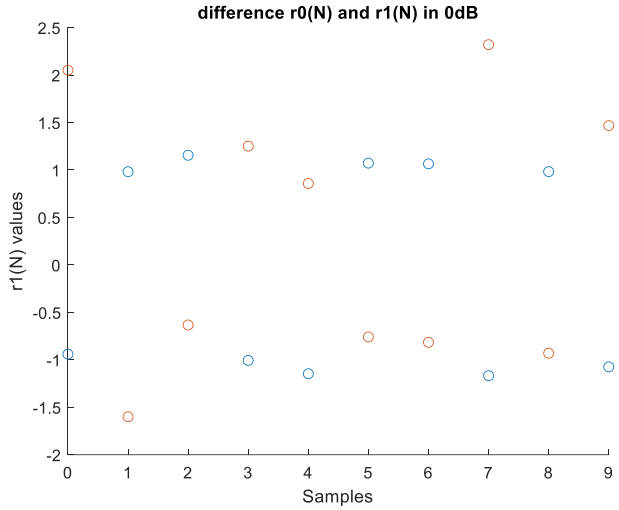
For -15dB→[-199.2157, 199.2157] →%99.21 error

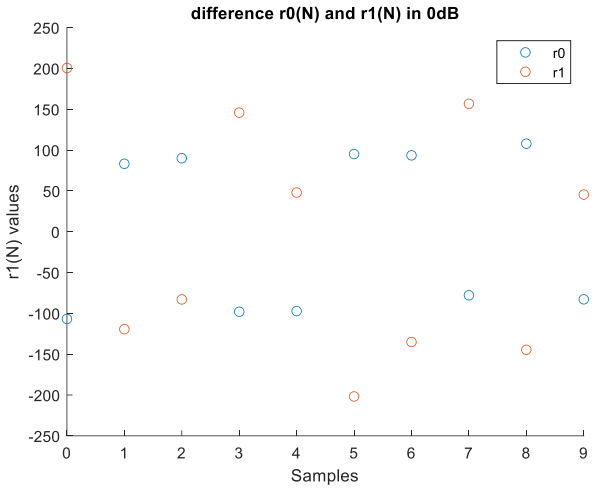
These rates more acceptable than r(t) rates for same SNRs (-15,0,15). But even high noisy signal (1:5.5) (-15dB SNR) the error is just %99 instead of %1907. This is the reason why they are called “optimum receivers”

3.5.c.1

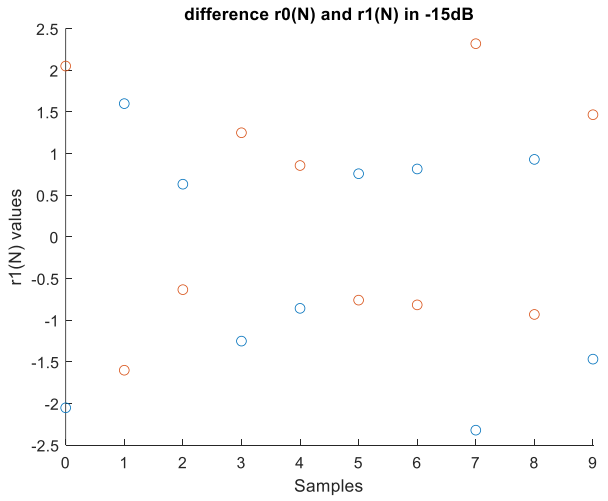


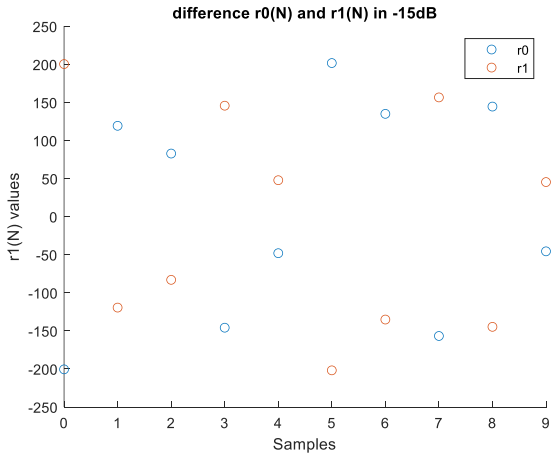
3.5.c.2


difference r0(N) and r1(N) in 0dB
r1(N) values
Samples


difference r0(N) and r1(N) in 0dB
r0
r1
r1(N) values
Samples

3.5.c.3


difference r0(N) and r1(N) in -15dB
r1(N) values
Samples


difference r0(N) and r1(N) in -15dB
r0
r1
r1(N) values
Samples

3.5.d

As we can see in graphs below as SNR values changes Pe changes exponentially (increasingly decreasing). It's reasonable such that while Signal to Noise Ratio is increasing, the error probability decreases because noise power becomes very little due to corresponding signal power so received is getting closer to what is transmitted while SNR is increasing. I have also made a mistake by giving SNR's in Amplitude form without multiplying by two taking square-root of each one. I have corrected this mistake. I have changed manual version of calculation of Probability of error. I thought that if we calculate both r0 and r1 for all the SNR values then compare each of them for which one is greater due to SNR values then save it to the array which is length of 310 (15+15+1)(SNR values)*10(Size of N) then If we substitute 100 from each of them and then taking mean of each 10 element length part (each 10 element part represents each SNR values) at result we get 1x31 vector which has same length with Q_function output. But the values are different. I didn't understand why these values are different.