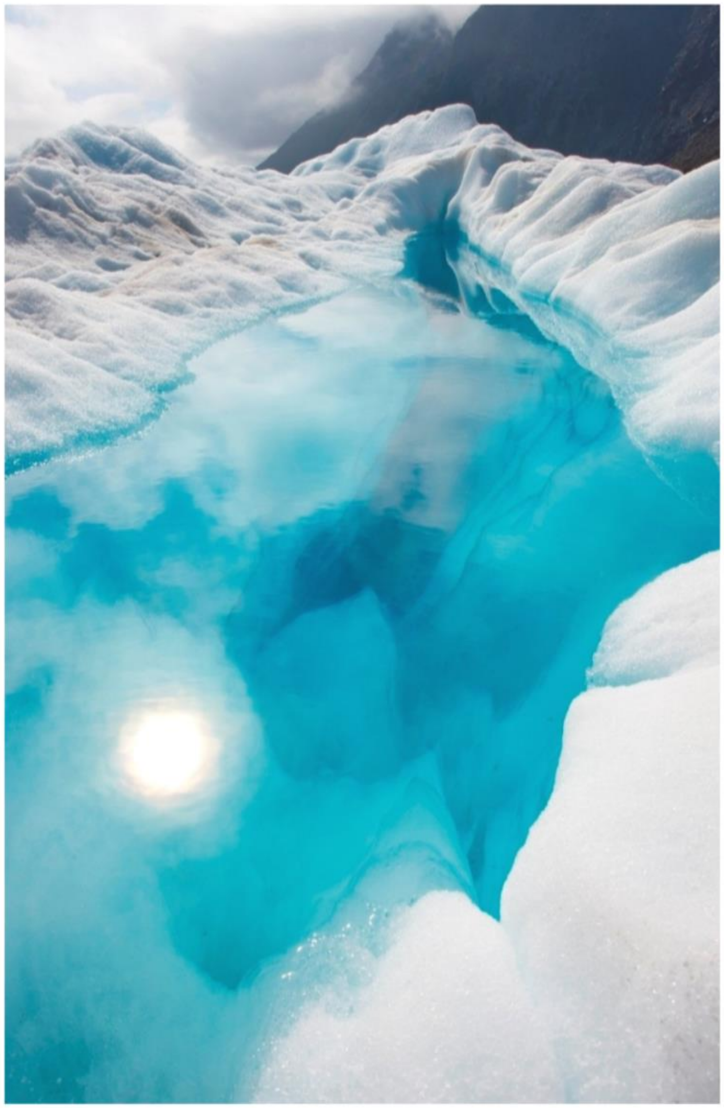


HW#6 Simulations

WIRELESS COMMUNICATIONS SYSTEMS

Fateme Noorzad | 810198271 | July, 2020

Part a:

This part is done with 10^4 bits.

For this number of bits, using 'randi' command, 0 and 1 are generated. These bits create a n*4 matrix, where each row represents a symbol block. Then each row is changed to decimal. Its decimal value represents what coded symbol should be sent.

Then these coded symbols are modulated. For QPSK modulation as was said in problem, $e^{jn\pi/2}$ is used where n belongs to {0, 1, 2, 3} set.

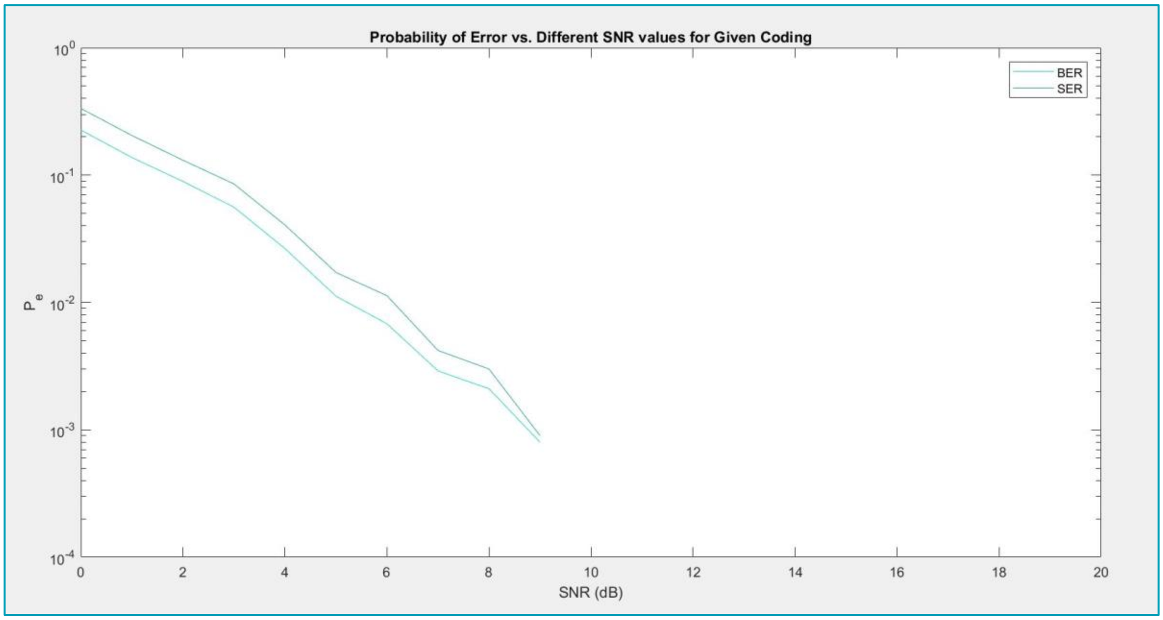
For simulating the fading channel, 'normrnd' is used. h is a complex normal random variable. Its mean is 0 and its variance is 1. So its real and imaginary part are normal random variable with 0 as their mean and ½ as their variance. All created symbols go through this fading channel. For simulating it, these symbols are multiplied to h. But since here we have a MIMO channel, in each iteration a 4*4 sub-channel of the original h is generated and multiplied to the symbols.

Then symbols which went through fading channel, go through an AWGN channel. For simulating this channel, since we want to plot P_e with respect to SNR, different SNRs should be considered. For this goal, AWGN noise is created with different variance representing different SNR values. This noise is also a complex normal random variable with 2/SNR variance and 0 as its mean. (We supposed symbols' energy is one) So noise has a real and imaginary part, each has 0 mean and 1/SNR variance. This noise is added to symbols.

Now it's time to detect the received symbols. For detecting we use maximum likelihood. If we represent received symbols with Y, the reference symbols with X (given code words in question.) and the fading channel with h, then we have:

$$X_{ML} = \min \left\{ \left| |Y - Xh| \right|^{۲} \right\}$$

After detecting symbols, probability of error should be found. So the detected bit matrix is subtracted from the sent bits and the number of non-zero indexes in this matrix is considered to be the errors. The fraction of these bits to all bits gives the probability of error. The resulting plot can be found below:



For finding the diversity order, the slope for large SNR is found. As a result:

```
Diversity order from performance curve is 5.228787
```

Although one point should be noted that, this diversity order varies between 6-8 in different runs.

Part b:

For this part in contrast to part a, symbols are sent using Alamouti scheme. Since 4 symbols are used and since QPSK is used, each of these symbols consists of 2 bits, the total number of states will be $2^{2*4} = 256$. For finding the coded symbols, first all numbers between 0 to 255 are changed to binary. The result is a 256*8 matrix, each row containing a number in binary form. Then in each row, 2 bits are changed to decimal in order to make a symbol. Then using the given equation for QPSK, modulated symbols are created.

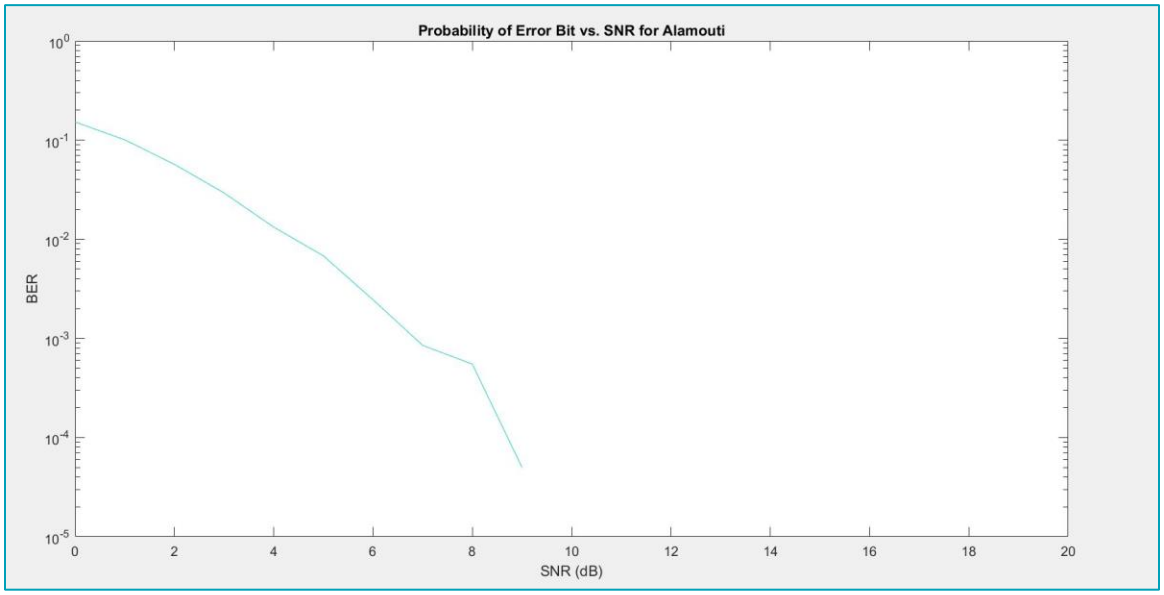
Since we are using Alamouti and based on the encoding given in problem, 2 different streams are sent. One is like part a and nothing is changed, the second one is conjucated and rotated as given in the problem. So these 2 are multiplied separately to h, to created faded symbols.

Same as last part, based on different values of SNR, variance of noise is calculated and for these 2 data streams, noise is added to symbols already went through fading channel.

Again same as last part maximum likelihood is used to detect symbols with one point in mind. Here since we have 2 streams of data norm for each is calculated and then added in order to show the true value of norm for the whole data stream.

After detection, only one other step remains which is to compare the detected symbols with the actual sent ones in order to determine BER. Since finding BER with detecting bits take a long time, symbols are detected, and then the probability of symbol error is divided by 2 in order to find BER.

The resulting plot can be seen below:



Based on the slope of curve in high values of SNR the diversity order is determined. So:

```
Diversity order from performance curve is 10.413927
```

Part c:

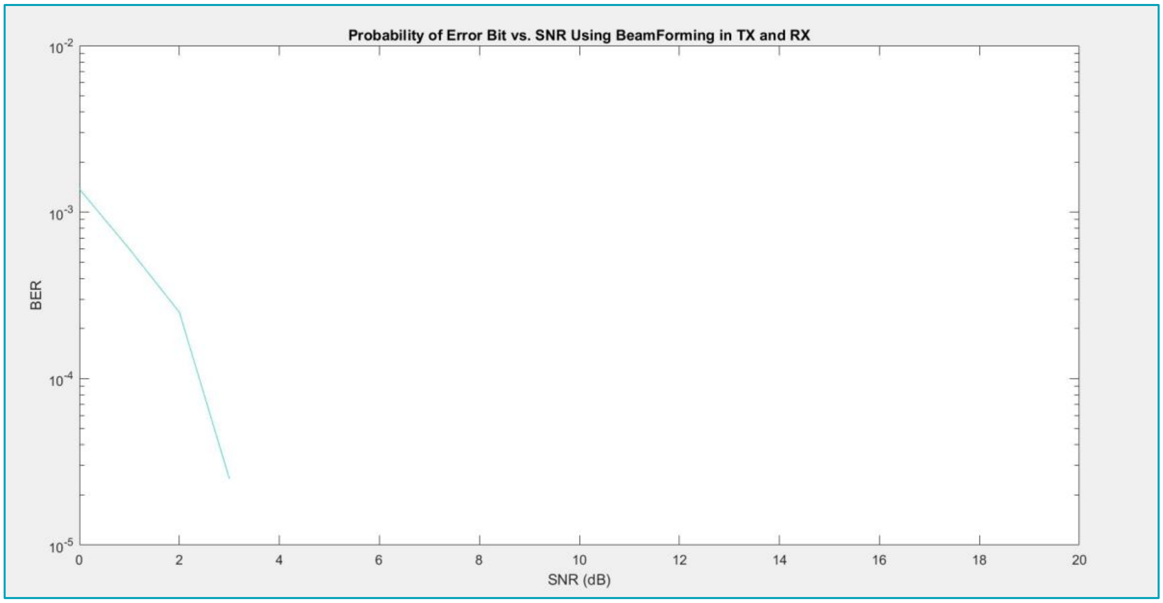
In this part, since 16-QAM is used, based on Gray Coding, place of each code is found and used as the modulated symbols. Using these references, the random generated bits are coded and sent.

With the same approach taken in part A and part B, h is simulated. But here in the problem it is asked to use TX and RX beamforming to get maximum diversity. So first h is decomposed using SVD. Since we want to get maximum diversity, maximum eigenvalue of each sub-channel of h is used.

Using the below equation the received symbols are simulated and are ready for detection.

$$Y = \Sigma s + U'n$$

Then same as the procedure carried out in the last 2 part, symbols are detected and probability of error is divided by 4, in order to find BER. The result can be seen in the below plot:



The diversity order is found as:

```
Diversity order from performance curve is 10.000000
```

Although the goal is 16 which can happen in some runs.

If we compare the BER of these 3 part with one another, the resulting plot is determined for the same bits and channel. As can be seen using beamforming to get the maximum diversity results in getting a better BER and also a better diversity as was expected, and part b is better than part a as well.