HW#5

SIMULATION REPORT

Fateme Noorzad | AFT | Azar 1398

Problem 1: (All these were explained in last HW)

First of all, the *4-QPSK* symbols need to be simulated. For this goal the signal size is set 1000. Meaning there are 4000 samples of sent symbols.

As been defined in the problem, all sent symbols are i.i.d. and have the same probability of happening. With that is mind, a random number is chosen between 0 and 1 uniformly. If the generated number is less than 0.25, the first symbols is sent. If the generated number is between 0.25 and 0.5, the second number is sent, and so on. In this way 1000 samples of sent symbols are generated.

Second step is to create vector $\underline{u}(i)$. This vector is the output of channel having 4000 columns (the number of samples of sent symbols). Based on these and impulse response of channel, $\underline{u}(i)$ is simulated. (The z-transform and finding one of the elements of this vector is written in HW.)

Third step is to create vector $\underline{x}(i)$. This vector is the result of adding a Gaussian complex noise to $\underline{u}(i)$. Real and imaginary parts of noise has Gaussian distribution with zero mean and variance 0.01. So a noise with 4000 samples is created and added to $\underline{u}(i)$ in order to make $\underline{x}(i)$.

Here if the loop is in training mode, training data is used. Meaning data coming to find error, is actually a shift of $s(i)$. The optimum shift (Δ) is found is the first part of the problem. Based on its value the desired signal is created. For samples with index less than Δ, desired signal is zero and for more than that is s(i).

Now let's continue with problem:

PART 1: SIMULATING LMS:

$\underline{u}(i)$ and $\underline{x}(i)$ are found as has been explained. In this part $\Delta = 17$, so based on this desired signal (d) is found and simulated.

LMS simulation:

LMS algorithm uses the bottom formula to update the weights of filter (here equalizer):

$$W(i+1) = W(i) + \mu e^*(i)x(i)$$

In the above formula, $\underline{W}(i)$ is a vector with size M*number of samples, containing all the founded weights. $e(i)$ is error which is :

$$e(i) = d(i) - y(i)$$

$$y(i) = W^H(i)x(i)$$

So based on the above formulas, y is going to be a number for each sample i making e be a number for each i^{th} sample. As a result W is a 35*1 vector.

To implement this algorithm $\underline{x}$ is cut in each iteration and then the above formula is used.

In this algorithm, in each iteration J, the expected value of error, is found. J is calculated using:

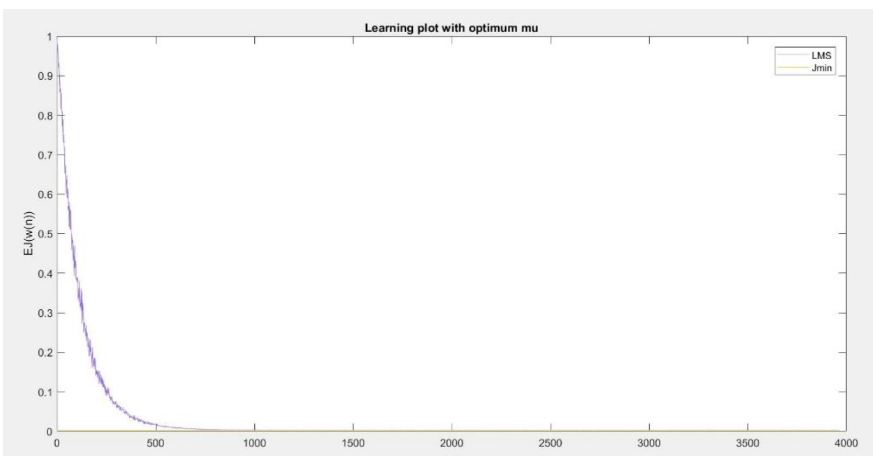
$$J\big(W(i)\big) = \ e * conj(e)$$

In this part the number of re-dos of iteration is said to be set 100. So *J(W(i))* are found for each i then all of the are summed up together and divided by 100. So actually *E{J(W)}* is calculated and shown as the output of algorithm.

For this part *M* is 10% :

$$\mu \cong \frac{٢ * ٠.١}{١٧٣.٢٥} = ١.١٥٤٤ * ١٠^{-٣}$$

For both of these values, the simulation is done. The results are:



Plot1. Using the approximated value of μ

```
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment is :10.106401
```

PART 2 : SIMULATING NLMS :

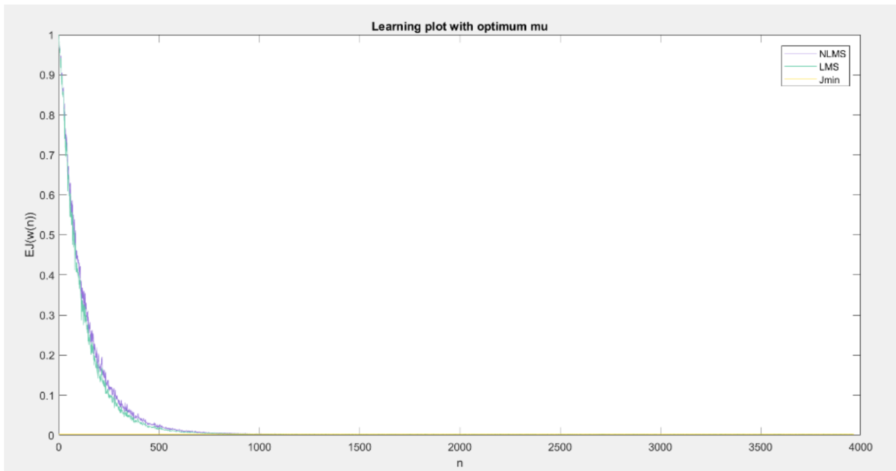
NLMS is same as LMS except that in updating w instead of

$$W(i + ١) = W(i) + \ \mu e^*(i)x(i)$$

Below formula is used:

$$W(i+۱) = W(i) + \frac{\mu e^{*}(i)x(i)}{||u(i)||^{۲}}$$

This factor is added to prevent the algorithm form diverging as a result of big inputs. Besides this fact, it is faster than LMS too. It can be seen from the below plot:



Plot2. Using the approximated value of μ

```
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment is :10.477913
```

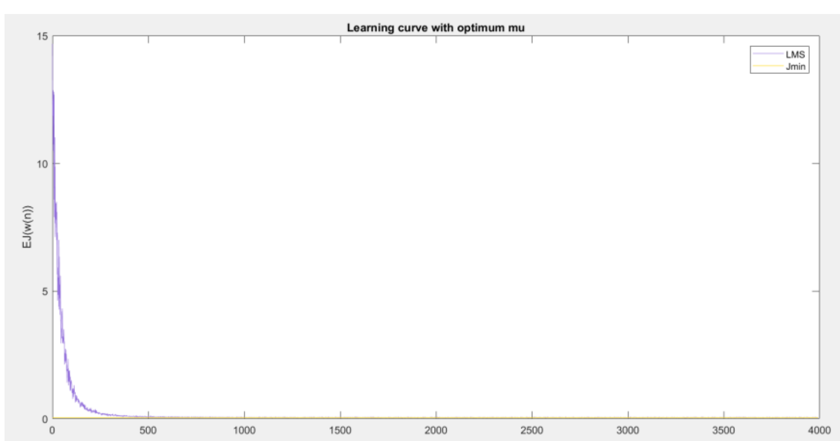
As can be seen LMS is slower in convergence, so its curve is under NLMS'.

Problem 2:

PART1: LMS SIMULATION:

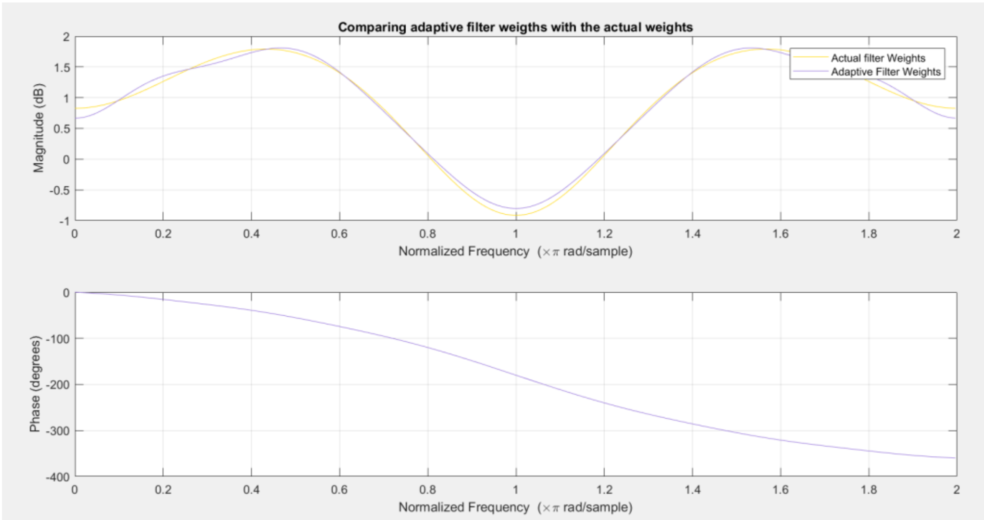
Same as the last part a LMS algorithm is implemented. (All calculations are done in paper and matlab.)

Learning cure will be:



Plot3. Using the approximated value of μ

After convergence happened, weights of the adaptive filter is compared with coefficients of W(t). The result is plotted in the below curve:



Plot4. Comparing calculated weights and the actual weights

As can be seen the calculated weights are pretty good and are really near the actual weights.

```
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment is :5.644415
With 4000 number of iterations (signal size) and 100 number of re-dos, spread value is :0.057608
```

PART2: DCT AND LMS COMPARISION:

In this part first DCT is implemented using the below formula :

$$T = [t_{ij}] : \ t_{ij} = \begin{cases} \frac{۱}{\sqrt{M}} . i = ۰ . j = ۰.۱. \dots . M - ۱ \\ \sqrt{\frac{۲}{M}} \cos\left(\frac{\pi(۲j + ۱)i}{۲ * M}\right) . i = ۱.۲. \dots . M - ۱ . j = ۰.۱. \dots . M - ۱ \end{cases}$$

Then it is used on the input vector, and after that error is found. The next part is about finding the approximate value of variance. With the approximated values, filter's weights are updated.

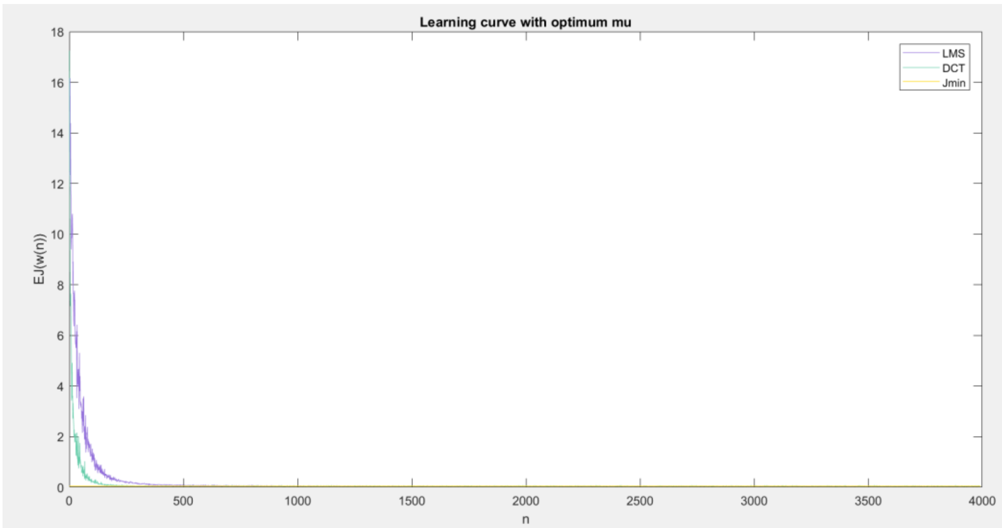
$$X(n) = Tu(n)$$

$$e(n) = d(n) - {W_T}^H X(n)$$

$$\sigma_{X_I}{}^{۲}(n) = \beta \sigma_{X_I}{}^{۲}(n - ۱) + (۱ - \beta)|x_i(n)|^{۲}$$

$$w_{Ti}(n+١) = w_{Ti}(n) + \mu \frac{١}{\sigma_{X_I}{}^{٢}(n)} e*(n)\, x_i(n)$$

Implementing the above formulas the learning cure will be:




```
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of LMS is :6.426276
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of DCT is :8.259968
With 4000 number of iterations (signal size) and 100 number of re-dos, spread value is :0.247860
```

PART 3: DST AND DFT:

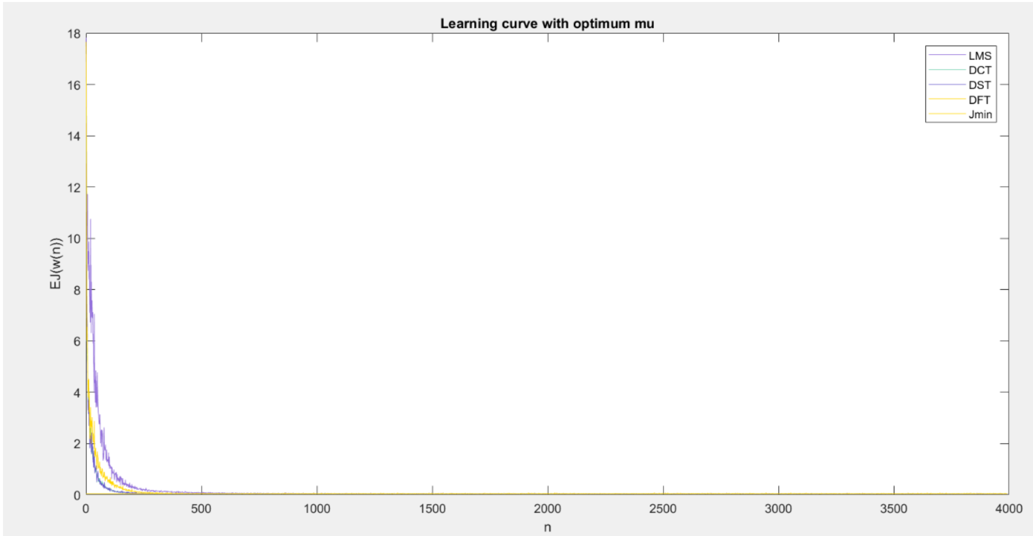
Algorithm implementation in both of these transforms are the same and the only difference is the transforms. For DST we have:

$$T = [t_{ij}] : \; t_{ij} = \sqrt{\frac{٢}{M+١}} \sin\left(\frac{\pi ij}{M+١}\right). i = ٠.١. \dots . M-١ . j = ٠.١. \dots . M-١$$

And for DFT the transform will be:

$$T = [t_{ij}] : \; t_{ij} = \sqrt{\frac{١}{M}} \exp\left(-j\frac{٢*\pi}{M}\right). i = ٠.١. \dots . M-١ . j = ٠.١. \dots . M-١$$

Results are shown as below:




```
With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of LMS is :5.649960
 With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of DCT is :7.334588
 With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of DST is :7.406444
 With 4000 number of iterations (signal size) and 100 number of re-dos, misadjustment of DFT is :7.083658
 With 4000 number of iterations (signal size) and 100 number of re-dos, spread value of DCT is :0.224939
 With 4000 number of iterations (signal size) and 100 number of re-dos, spread value of DST is :0.232744
 With 4000 number of iterations (signal size) and 100 number of re-dos, spread value of DFT is :0.200960
```

As the above results show, LMS has the best miss adjustment, but the other 3 has better spread values.