Real-Time Digital Signal Processing
Prof. Rathna G N
Department of Electrical Engineering
Indian Institute of Science - Bengaluru

Lecture - 34

Lab: LMS Algorith in MATLAB

Welcome back to real time digital signal processing lab this time. So, we will see that how we will be implementing least mean square algorithm that is LMS algorithm in MATLAB today we will look at it. So, this is the as I will be mentioning usually we will be taking asking the students to implement it. So, this is one of the student who has implemented. So, which I am taking it for your presentation. So, here this is going to call a GUI function basically as it is easier to run with different input signals.

So, we will see how our LMS algorithm is going to work. So, we are going to some of the name and structure you will be seeing it GUI definition. So, those who are interested in writing their own GUI you can refer to this example. So, we are interested in seeing that how our LMS algorithm is going to work.

So, we are going to call the variables here as you can see, we have the declaration because for parameters changing they have to be declared global here also x, y and then y out have been declared as global and the order of the filter is 20 in this case. So, you can vary this order and then see how your LMS algorithm is going to work. So, now, what you have to do it. So, you are going to weight function w initially you are making them zeros. And then in this case mu is selected as that is time step is selected as 0.

006. So, we will see first we will see what happens with the 0.006 and then later on we can see by modifying it step size varying it if you have not calculated it properly what is the thing is going to happen. And then for the length of this thing filter minus the order what will be taking it that n plus l minus 1 what we have it. So, buffer is going to be configured.

So, this is going to be order i plus order minus 1 because you are starting from 1 and then you will be taking the transpose basically and then y out of i is nothing but. So, you have buffer star w star 2 what you will be calling it. Then error i is going to be yi minus buffer of w whatever value is the thing. So, this is the updated what we are going to have it as error. And then our weight function w here it is not represented as n plus 1 w is equal to the current value of w plus you will be calculating you will be seeing that whatever the value here into mu into error of i which you will be taking it as transpose that is update the weights basically and then you will be ending the code.

So, then you will be plotting it. So, we will run it has all the 3 algorithms that is NLMS that is normalized least mean square algorithm and RLS algorithm is also there. So, I have considered here only LMS for this class. So, we will see. NLMS and then RLS how the function is going to be used and will as I mentioned in the class will not be deriving the equations for it those who are interested can run the code and then check it ok.

So, we will run this code you will be seeing that LMS algorithm is going to come up. As I have mentioned so, some of the laptops will have resolution problem. So, we will see that we will be running this code as LMS algorithm first and then we will see the thing. So, we will run the code this is going to give us as you are seeing it adaptive filters both LMS, NLMS and as well as RLS has been integrated in this and in the next class we will see separately how we are going to design the thing. First is I will be loading my noisy signal.

So, this is my noisy signal what I will be loading on to the system and then what desired signal I want to in this case I know both of them. So, I will be loading it. So, open it and then we can run our LMS algorithm. So, both I have loaded. So, this is the original signal and this is the noisy signal what you are seeing it and using the least mean square algorithm with mu set to 0.

006 that is the optimum what usually found out by trial and error you can do it or you can exactly calculate. So, we will see in the class how we are going to calculate that. So, we will be getting back approximately our original signal as you can see although amplitude has got a little bit modified compared to the original one. So, this is what we have taken is input is the signal. Remember the force will be with you always.

something must be striking in your mind we use the same input for our FIR and IR filter. So, same thing we are using for our LMS algorithm also. So, we will see the noise here whatever noise was added in using the MATLAB which was stored as the noises signal in this case. as you can see single frequency tone was added. So, if you want you can add any noise and then if you have your desired signal you can this is after running our LMS algorithm that is filter output.

So, we can play. So, you can see that your noise got eliminated. So, this is how one can run the thing with whatever mu selection what we have done. So, we will see that if mu is equal to 1 what is the thing is going to happen. So, we will run this case and we will see whether we are going to get back the signal or it is going to have some noisy signal present in the thing. This is the desired signal what I have to load it and then run the LMS algorithm.

you can see that my input step size is 1 ok. So, you are seeing that if you play the thing nothing what you are going to get it. So, you will be seeing how the steepest algorithm is

going to work. Now, we will go back and then change our thing and then rerun. So, this was the maximum one I have taken the thing.

So, we will start from point 1 and what is the minimum what we are going to get we will see it. So, if it is going to work for 0.1 this is the trial and error method what I am using it, but how we are going to calculate as we have mentioned that the calculation of mu has to be using the if you are calculating on the go then we have to do the inverse of the matrix or I can calculate using the are coefficients basically you have to do matrix inversion this is a noisy signal. So, we will be loading the desired signal again and then I can run the LMS algorithm. So, you will be seeing that most of the thing you have got back with.

Remember the force will be with you. how your steps size has to be one has to calculate. So, it depends on how much iterative algorithm you are going to run. So, as you have seen the thing although you are getting the. Remember the force will be with you always.

So, this is what the output is going to be. What was the reason for 0.006? How slow I can go that is what we will be looking at it. So, whether you are going to hit the minimum or not. So, I will be taking so many 0s and 1s.

So, still we can run this code. Again we load them and then we will run our LMS algorithm. So, you will be seeing that my step size is too low in this case. So, this is what the your plot gives you and then if I play what you will hear you can Can you guess it? So, you are what is it audio is completely reduced or the speech has got very I do not know whether you have heard it or not here we can hear it very. minute value under that amplitude what we are hearing.

So, there is a single tone sine wave also which is going. So, you have to compute your mu what should what is going to give you your minimum value. So, if you want to plot your weight function, so you can plot them also. So, we will see by will see plot w because I am touching in this thing. What is it? gy, I may get error.

So, we will load it again. I think my plot is outside. So, that is why it is not shown with the thing. Let me see by taking it inside whether it will be giving me the thing. Otherwise just without GUI we will run in the next class our LMS algorithm. So, there will be error in the thing because if I come out of it, it is going to it is not plotting basically and if I am inside you can see that it is going for the plot and then it is not coming out of the handle.

So, we can reduce to 0.001 and then look at it how this is going to have the impact on our output. There was error in opening the GUI, it took long time to open it. So, we will clear all and we will open the function again. We will run the thing. So, our load noise is signal and then we load the desired signal.

and then we will run our LMS algorithm. So, you will be seeing that compared to the original with 0.001. So, you will be seeing that there are attenuation happening with the thing.

So, when you play the sound. Remember the force will be with you always. Remember the force will be with you always. So, it is a little bit working on the thing. So, what usually you have to find out what is optimum you what you can run with the thing. So, thank you we will see separately running and then how the weight function is going to adapt itself in the next class. Thank you.