Performance Comparison of Linear FIR filter vs IIR filter for Adaptive Echo Cancellation

Mukhil Azhagan Mallaiyan Sathiaseelan
Electrical and Computer Engineering
University of Florida
Gainesville, US
mukhil.mallaiyan@ufl.edu

***Abstract*—In this paper we will quantify the performance of Linear FIR filters and compare its performance to the gamma filter. We will introduce both finite impulse response (FIR) filters and Infinite impulse response (IIR) filters, and more specifically a variant called the Gamma filter, show how the individual filters work by varying the parameters for its models, and then compare their performance with the results obtained. The filters will be applied on an Echo cancellation problem to adaptively remove the echo from the transmitted signal during a phone call. Graphs of various metrics such as Echo return loss enhancement (ERLE), weight tracks and learning curves are presented to compare and quantify the performance and show the superiority of gamma filters over FIR filters when presented with a non-stationary corrupted signal.**

Keywords—Echo cancellation, Adaline, gamma filter.

I. Introduction

The Adaptive Linear system is a versatile tool in machine learning. The simplicity of the linear model is a welcoming aspect that allows its use in a variety of scenario, in cases even where the underlying model is non-linear. These have been and are being used in various spheres for prediction, echo cancellation, interference cancellation, classification, signal enhancement and system control.

The use Finite Impulse Response (FIR) filters for adaptive signal processing is widespread. Within FIR filters various methods can be applied to adapt the system. The Weiner Analysis of the Weiner solution for batch learning and the Least mean square (LMS) approach for on-line learning are some of the popular methods that are used and thus, these will be of interest to us in the discussions that follow. Amongst Weiner analysis and LMS, we will discuss how the LMS approach is better for nonstationary signals, though only for slowly varying ones.

An Infinite Impulse Response (IIR) called gamma filter [1] will be used for the same echo cancellation problem and a noticeable performance increase will be visible. The recovered signal, as will be discussed in *Section II*, will also be much more intelligible.

The paper is organized in the following manner. *Section II* will discuss the methodology and the working of the filters. *Section III* will present results and plots for various parameters and filter orders. *Section IV* will be about a discussion on the performance comparison among the two filters. *Section V* will be the conclusion.

II. Methodology

We are dealing with an echo cancellation example (see *figure 1*) where the goal is to adaptively remove the echo, that is, the signal from Caller A that goes to Caller B and returns with a delay. It is to be assumed that the hybrid at the far end (At caller B) is working fine. This interference has corrupted the signal reaching the far end. What we are trying to achieve is to adaptively cancel the echo and try to recover the signal using two means, one through an FIR filter and another through a gamma filter

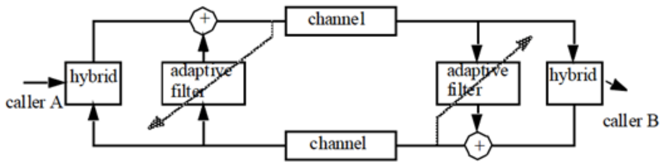


Figure 1: Model of the problem

As discussed, the premise for the paper is that , speech from caller A is corrupted and only the loud music from A is heard at B .we have with us two datasets , one for the corrupted speech and one with the music . We have to work with these datasets to adaptively restore the signal . This becomes a case of Supervised learning . We will discuss the methodology below . Detailed explanation is available at [6].

A. *Model for Supervised Learning*

An adaptive system is one that can modify its parameters based on a feedback to obtain an optimum solution. As can be seen in *figure 2*, the design of an adaptive system for supervised learning is presented[2]. The important parts of an adaptive system are its cost function, its modelling function or the mapping function (In this case it is linear) and a training algorithm that trains the modelling function based on the cost function.

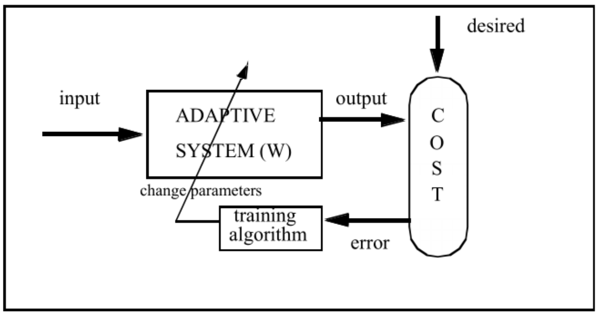


Figure 2: Model of an Adaptive system

When given a set of samples $\{x_i, d_i\}_{i=1}^{N}$, where x_i is the i^{th} input sample and d_i is the i^{th} desired sample. We create a linear mapper defined by y.

$$y(i) = \sum_{k=1}^{M} w_k x(i-k) + b$$

For filters to maintain linearity, we omit the bias *b*. So, the equation becomes,

$$y(i) = \sum_{k=1}^{M} w_k x(i-k) \qquad \rightarrow \textit{Eq. 1}$$

The error is defined by,

$$\mathrm{e(i)} = \mathrm{d(i)} - y(i) = d(i) - \sum_{k=1}^{M} w_k x(i-k) \qquad \rightarrow \textit{Eq. 2}$$

Our cost function is defined by,

$$\mathrm{J(w)} = \frac{1}{2N}\sum_{i=1}^{N} e_i^2 \qquad \rightarrow \textit{Eq. 3}$$

Expanding it, we obtain

$$\mathrm{J(w)} = \frac{1}{2N}\sum_{i=1}^{N}\left(\mathrm{d(i)} - \sum_{k=1}^{M} w_k x(i-k)\right)^2$$

To find the optimal solution, we need to minimize the weights parameter given by *w* so, we solve,

$$\frac{\partial J(w)}{\partial w} = 0$$

This is the methodology used to model the adaptive system, we will discuss this methodology for two different filters in the upcoming section. For both filter types, the desired signal is taken as the corrupted signal and the input signal is taken as the music. The error will give the recovered speech

B. FIR Filters

1) LMS approach.

The strength of the LMS algorithm lies in the fact that, it works on samples on a one sample at a time basis. Due to this, it can track the changes in the sample set much more effectively than the Wiener method, which only takes an average of the weights for the sample set.

The weights are varied continuously in every iteration, that is, for every sample. So, even if the weights start from a value of 0, it can optimize itself over time. The equation that governs this update is given by,

$$w_k(i+1) = w_k(i) - \mu\frac{\partial J(w)}{\partial w_k} = w_k(i) + 2\mu e(i)x(i) \rightarrow \textit{Eq4}$$

Where μ is the step size? Optimizing μ is by itself an experiment in itself. While there are formulae to compute it, the most common of which being that μ being twice reciprocal of the maximum eigenvalue of the input autocorrelation matrix. Here we will take a set of values for μ and discuss the results. In some cases, to ensure that the weights are not deviated too much by those values that have a large variance, we find it fit to modify *Eq.4* and normalize it.

$$w_k(i+1) = w_k(i) + 2\mu e(i)\frac{x(i)}{x^T(i)x(i)} \qquad \rightarrow \textit{Eq.5}$$

We will use *Eq.5* in our trails.

C. IIR Filter

An M[th] order discrete-time finite impulse response (FIR) filter is only able to *remember* M+1 samples. In cases where a larger memory is required, we need to employ such filters as an infinite impulse response filter (IIR), to consider values farther away in the past. As will be evident from the Gamma filter[1] we will now discuss.

As one would expect due to the infinite response of the IIR, the question of system stability would arise, but this is taken care of by the gamma filter using its own parameter μ.

The gamma filter is defined in a similar way to *Figure 2, with* the mapping function y defined by,

$$y(n) = \sum_{k=0}^{M} w_k x_k(n) \qquad \rightarrow \textit{Eq.6}$$

Where x is defined by,

$$x_k(n) = (1-\mu)x_k(n-1) + \mu x_{k-1}(n-1) \qquad \rightarrow \textit{Eq .7}$$
$$where\ k = 1, \ldots, M$$

This is a single stage gamma filter. There can be multi stage gamma filters as well. The design of a gamma filter is discussed in (1). Since it has parameters as weights and μ ,the filter is also called Adaline(μ) or adaptive gamma filter.

We will however be concerned about μ , the parameter that governs the stability of the gamma filter. This value of μ should satisfy $0 < \mu < 1$. For the purposes of the paper, we will deal with μ = 0.2.

The weight update function will follow *Eq.5*. We will name step size as η .

Using the methodologies explained above, we will now look at results from Matlab® programs. The code for which is also attached with the paper.

III. Results

A. Plots of Training Data

The training data for this paper is a music signal and a corrupted signal that has the speech to be recovered mixed with the music signal . It is vital for us to analyze the plot of these signals to learn about jump discontinuities in the signal for us to better understand some ERLE and weight track plots that will be discussed .

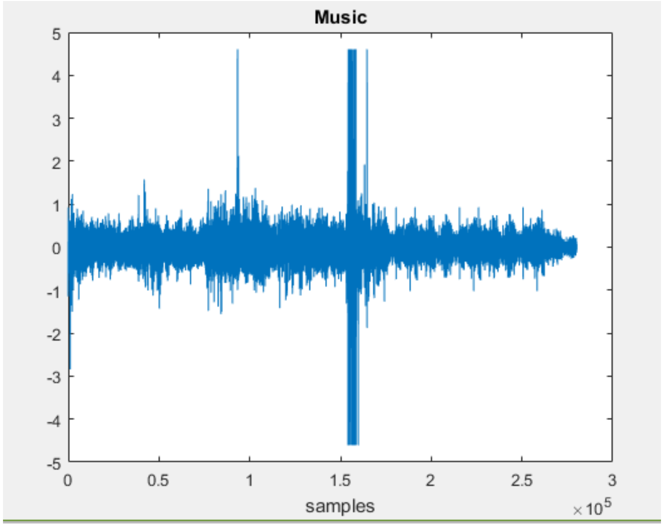


Figure 3: Plot of the music

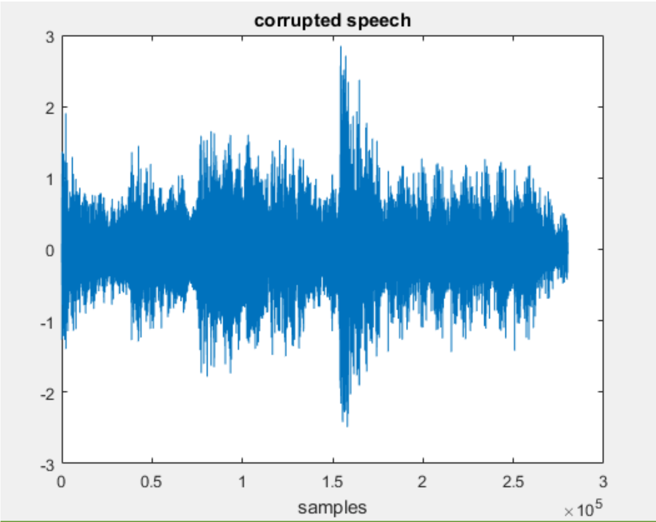


Figure 4: Plot of Corrupted signal

B. *Performance of FIR filter*

This is done using the LMS algorithm.

1) Ques A.1 . ERLE vs M

The step sizes are taken as [0.01,0.005,0.0025,0.001,0.0001]

The ERLE (Echo return loss enhancement) is given by,

$$ERLE = 10\log_{10}\left(\frac{\frac{1}{N}\sum_{i=1}^{N} d_i^2}{\frac{1}{N}\sum_{i=1}^{N} e_i^2}\right) = 10\log_{10}\left(\frac{\sum_{i=1}^{N} d_i^2}{\sum_{i=1}^{N} e_i^2}\right) \rightarrow Eq\ .8$$

Using the Eq.8, we can apply LMS. The plots are presented below

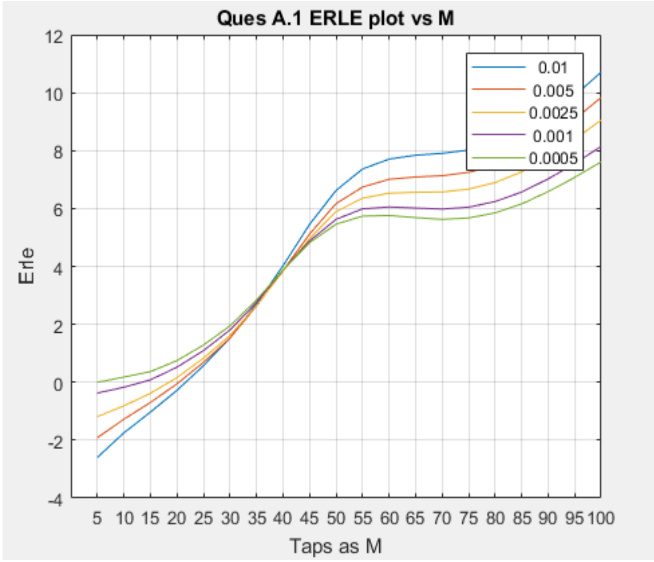


Figure 5: Plot of ERLE vs M for various step size

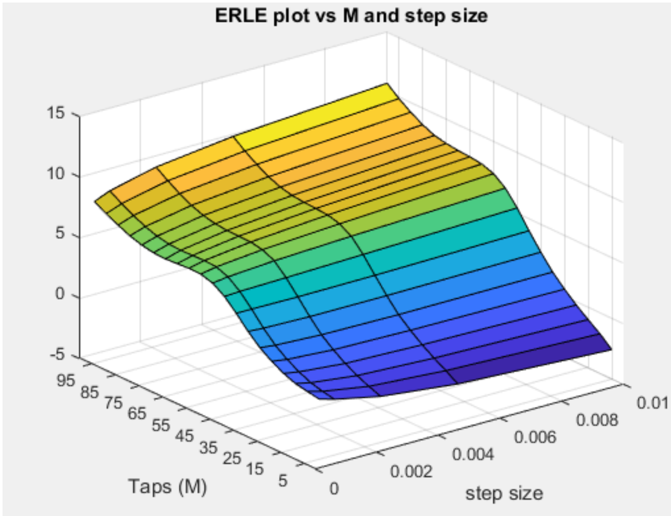


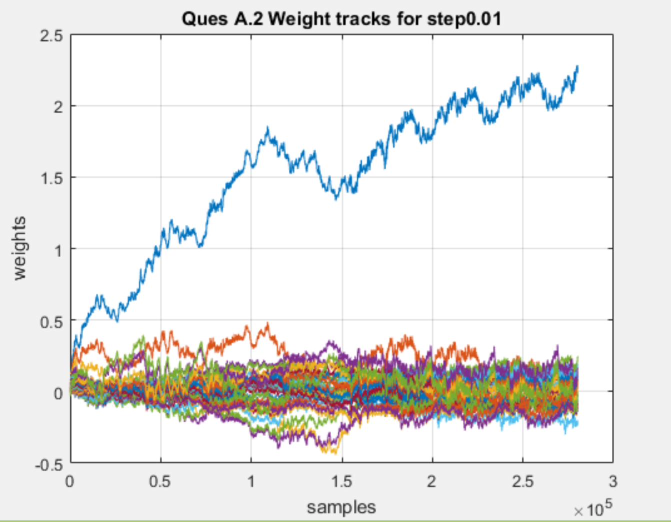
Figure 6: 3-D surface plot of ERLE vs M and step size

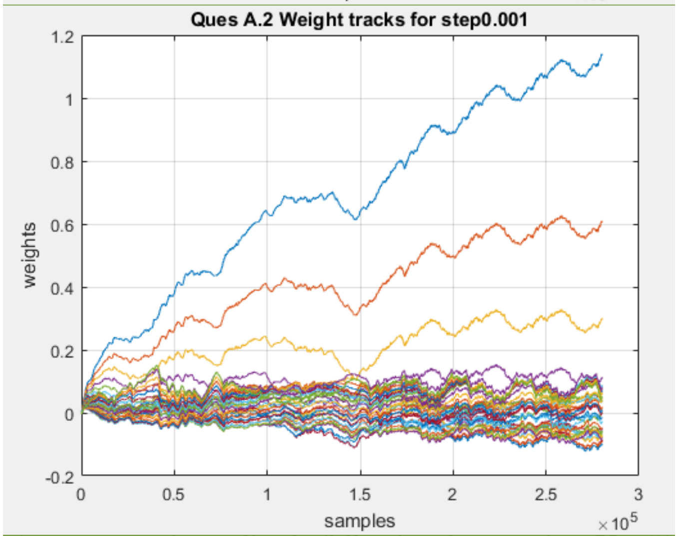
The 3-D Plot is also given for reference. Further discussion of the plots will take place in section IV. For now, from *figure 3*. It is evident that for taps of 40. We have a decent ERLE and it is independent of the step size. We will take M=40 for comparison of various step size and weight tracks

2) Ques A.2 . Weight Tracks

Weight tracks are presented for step size μ of [0.01 0.001 0.0001 0.00001] and for Filter order 40. This is arranged in a scale of 10, to cover as wide a step size as possible within a limited number of plots.

Due to limited space available, the plots do not have a legend. either way, it does not matter if one remembers that a weight track that varies quicker will correspond to a larger eigen value of the input correlation matrix and vice versa.





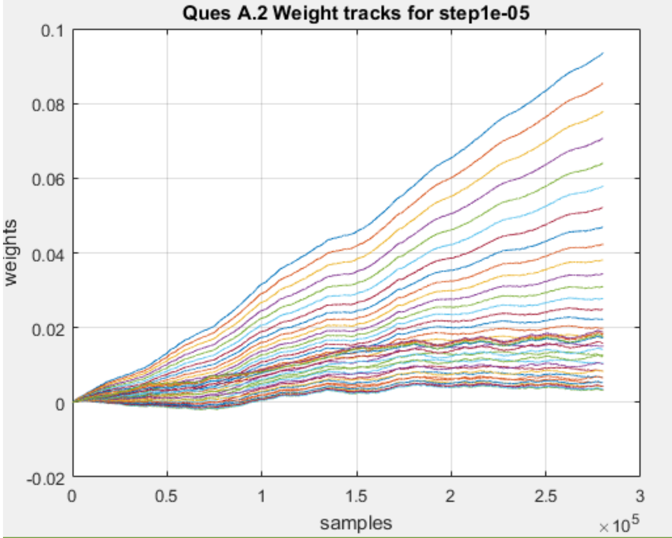


Figure 7: Plots of weight tracks across various step sizes

An important characteristic for weight tracks is the compromise between the speed of convergence and rattling. While for shorter step sizes such as μ=0.01 . There is a lot of oscillations and There is also a noticeable divergence from optima, why this happens will be in Section IV. for μ=0.0001 and μ=0.00001 it is a much smoother curve. But they do not converge to their optima within the sample length. So, we pick a size between 0.001 and 0.0001. Through multiple trails, it can be concluded that 0.0008 is a satisfactory μ . To be noted, the value may not be mathematically accurate and it is governed by various specifics.

Given below is a plot for 5 epochs, M=300 and μ=0.0008 and updating the final weights as the initial weights for every epoch. we can see it converging the graph beautifully. The sound is also attached, we obtain the recovered speech. It should also be noted that if step size is made smaller and epoch are increased many times. It is possible to get better recovered speech, but it is computationally expensive

C. IIR Filter -Gamma Filter

1) Ques B.1 . ERLE vs M

The IIR is run for filter order of 2 to 40 and μ=0.2 .

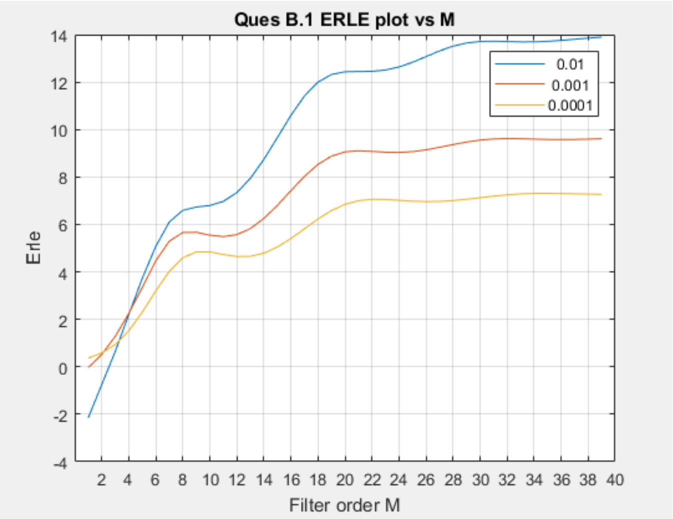


Figure 8: Plot of ERLE vs M for different step size.

There is a saturation in ERLE values after 30. For this reason, other plots for IIR will be taken till M=30. The noteworthy thing about Gamma filter is that it recovers the speech to a highly intelligible level. When step size η is taken as a comparatively large value such as 0.01. The system cancels echo quicker, while for smaller η the convergence is slower but more precise, though not visible in the ERLE curve, it will be evident when the recovered speech is played. The 3-D surface plot of ERLE vs M and step size η is also provided for reference.

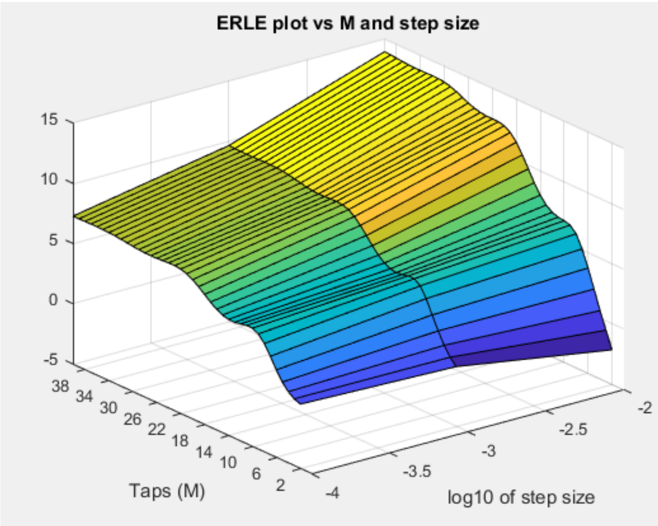


Figure 9: 3-D plot of ERLE vs M and log10 step size

2) Ques B.2 . ERLE vs μ

Fixing M=30, which gives the highest ERLE and step size 0.01 as from the previous subsection. We vary the feedback parameter μ.

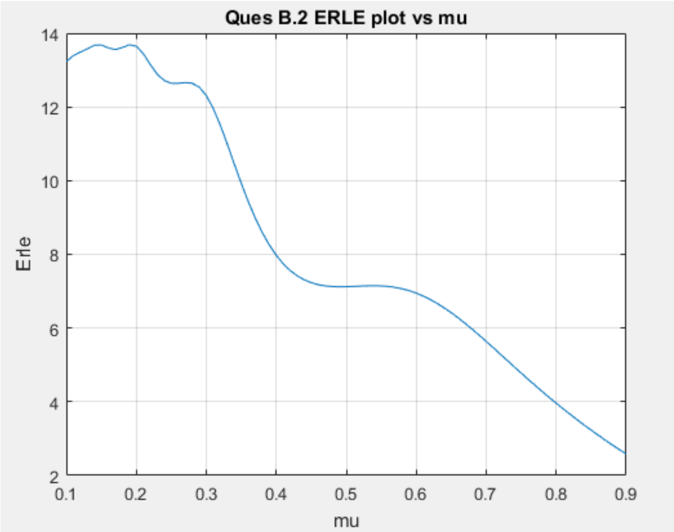


Figure 10. Plot of ERLE vs μ.

As is expected for increasing μ, the ERLE is decreasing. For better performance the IIR should have a converging response.

The 3-D surface plot of ERLE vs μ and stepsize η is also provided for reference.

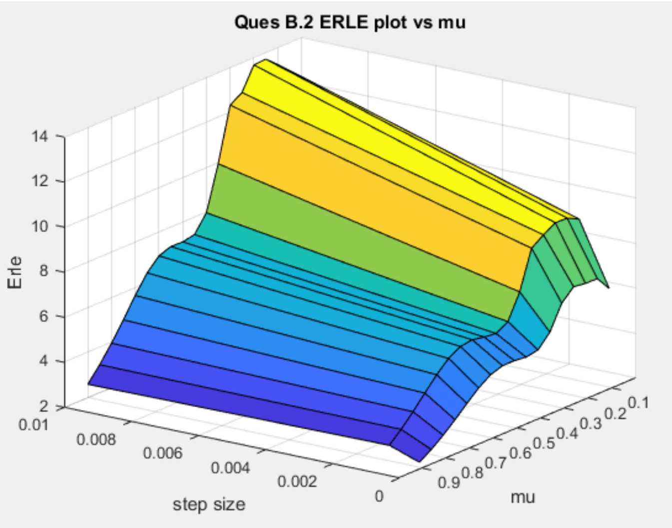


Figure 11. 3-D Plot of ERLE vs μ and η

IV. Discussion

The performance of FIR and IIR filters have been presented in the results section. The audio file attached to the project can also be listened to for audible difference in performance. It should be noted that, for FIR the audio file has been obtained after running 5 epochs, M=300 and μ=0.0008, while the audio file for the IIR is obtained with just 1 epoch for a step size of 0.01.

For the FIR filter, the LMS algorithm is used. The Wiener solution will not work for this case, because the signal is non-stationary. Even LMS faces problems due to its irregularity. Notice there are a few jump discontinuities in *Figure 1*. Due to this the weights are struggling to adapt, and for higher values of step size, it even starts to diverge at some points such as in

μ=0.01 . Why we do not take a step size governed by the formula $\mu < \frac{2}{maximum\ eigen\ value}$ is that even if μ = 1e-06, which is the neighborhood of maximum eigen value of the input data for various model orders, Due to number of samples and the presence of these discontinuities, it affects the model to a large extent that there are such unexpected and unwanted results.

This is overcome in the IIR because it has the influence of past data as well and it quickly converges back to optimal weights even in the presence of such jumps.

Another thing to note about FIR vs IIR are the learning curves. Presented below are learning curves

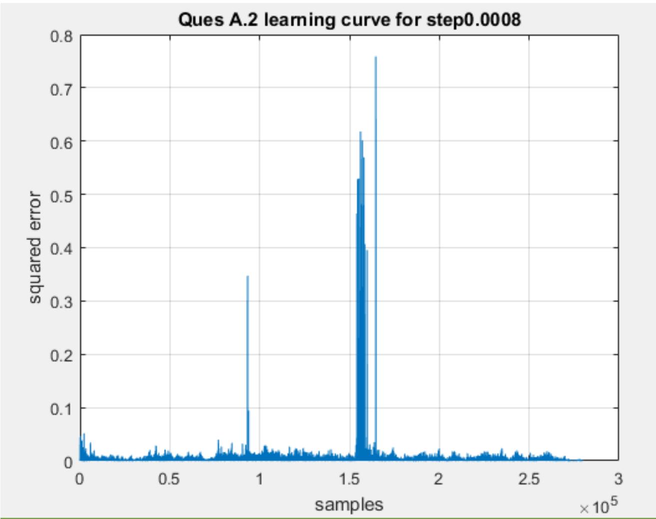


Figure 12: Squared error for each sample vs samples for FIR Filter

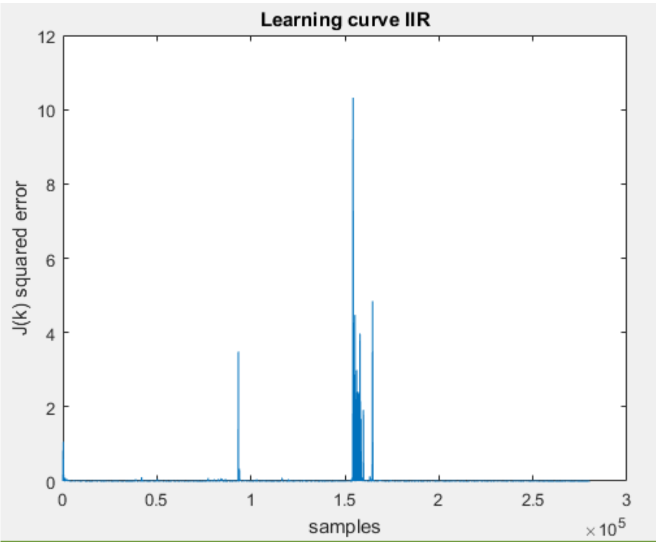


Figure 13: Squared error for each sample vs samples for IIR Filter.

As is evident from the plots there is the presence of an error at all times in FIR filter, Whereas in IIR filter (step size=0.01) it is close to none , apart from the short time during the jumps , it converges rather quickly .

Another interesting phenomenon noticed in the ERLE plots, for decreasing step size, the filter performs poorly. This is again due to the lack of data for adaptation. This should be solved simply by running for multiple epochs. Proof of this is provided in the audio file attached. Though intelligible speech cannot be recovered for the given problem in 1 epoch, we are able to get a much better and more intelligible speech for 5 epochs.

Things to be noted

Ques A.3:

Like mentioned in the previous paragraph, for a wide step sizes and wide filter order, the intelligibility is very low for FIR filters even with the LMS approach. When tested with M=40 and step size =0.01 ,the signal is mostly only loud music to the ears , with a small murmur of speech at the end . This, I believe, is due to the properties of LMS itself as it has had close to 200000 samples to train itself. For M=300 or some value in this neighborhood, it is comparatively better, but far from perfect.

However, for multiple epochs. The attached audio file for instance has 5 epochs, M=300 and μ=0.0008. The speech is recovered to a satisfactory extent.

Ques B.3

Taking similar parameters for IIR, M=30 (M=40 has saturated, so it has no effect) and step size =0.01 . The speech is recovered to a high extent. It is even possible to make out almost all the words in the speech. This is the superiority of the IIR over FIR filters

Overall performance is better for IIR filters.

V. Conclusion

The use of IIR filters, particularly the Gamma filter that is discussed here, for cases of non-stationary signal echo cancellation has shown much better performance over FIR filters. There are however other filters for the purpose. But in Linear Adaptive filters, the gamma filter has a good balance of stability, linearity and memory .
Linear models are praised for their simplicity, however, there are some cases where linear models might not work , It is necessary in such cases to use other nonlinear adaptive filters.

The Project however was a good review of the effects and limitations of a real-life example. One could think of various spheres of daily life to apply the gathered knowledge on what filter to use and when to employ such a filter. My personal preference would be to gather weather data and try to apply adaptive learning to it. The possibilities are many.

Acknowledgment

I confirm that this assignment is my own work, is not copied from any other person's work (published or unpublished), and

has not been previously submitted for assessment either at University of Florida or elsewhere.

Signature:
MUKHIL AZHAGAN MALLAIYAN SATHIASEELAN
Date:
10/09/2017

References

[1] Principe, Jose C., Bert De Vries, and Pedro Guedes De Oliveira. “The gamma-filter-a new class of adaptive IIR filters with restricted feedback.” IEEE transactions on signal processing 41.2 (1993): 649-656.)

[2] J. C. Principe, N. R. Euliano, and W. C. Lefebvre, “Neural and AdaptiveSystems: Fundamentals Through Simulation.” Wiley: Hoboken, NJ, 2000.I.S. Jacobs and C.P. Bean

[3] Christopher Bishop,Pattern Recognition and Machine Learning, Springer 2006.

[4] “Lecture notes and Info session,” EEL 5840,Elements of Machine Intelligence , University of Florida.

[5] Bernard, Widrow, and D. Stearns Samuel. "Adaptive signal processing." Englewood Cliffs, NJ, Prentice-Hall, Inc 1 (1985): 491.

[6] https://ufl.instructure.com/courses/343047/files/folder/Assignments/project1?preview=34401412