**Comparison of LMS and Gamma Algorithms for Echo Cancellation**

Yebowen Hu

Department of Electical and Computer Engineering, University of Florida

**Abstract**—*This project is to design a finite impulse response(FIR) filter and an infinite impulse response(IIR) filter to filter out speech signals from a corrupted speech audio signals. Two inputs are desired corrupted speech and music sounds. FIR filter is designed by implementing least mean square(LMS) algorithm while the IIR filter is designed by implementing Gamma algorithm. In this paper, step-size and model order will be decided due to the quality of outputs sound. Mean Square Error(MSE) and Echo Return Loss Enhancement(ERLE) will be used for performance evaluation.*

Keyword- adaptive FIR filter & IIR filter, LMS, Gamma, Performance Comparison.

# Introduction

This paper is about comparison of two different filter performance for echo cancellation. For solving this echo cancellation problem, corrupted speech signals are seemed as desired signal and the input signal for training is music signals. So, during the process when we filter out the music from corrupted speech, the error signal which has been extracted is the speech signals that I need. For the best performance of filter, the filer order, training step size should be determined at first. Then I compare those error signals by listening and comparing the value of ERLE.

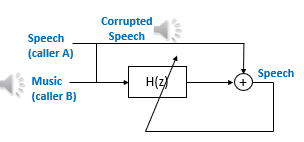


Figure 1 The basic map for interference canceling adaptive filter

# Methodology

Adaptive filter is one kind of model based supervised learning method. A set of input samples are given as:

filter weights according to the current iteration so that the estimation index can be reduced in the next iteration. To cancel the interference, a noise N1 has to be compared with a desired signal which consists of the speech signal S and another noise N0.

The cleaner output signal e is going to be

e(n)=S(n)+N0(n)-w(n)\*N1(n) (2.1)

## Least Mean Square-LMS

LMS is a steepest descent algorithm. It uses a simple estimate of the performance surface.

 (2.2)

The recursion formula for the LMS algorithm is

w(n+1)=w(n)+μ‘\* (d(n)-y(n))\*x(n) (2.3)

μ‘=2μ,

in which μ is the learning rate and μ‘ is the step-size.

The algorithm will converge when 1<μ<1/λmax, where λmax is the largest eigenvalue of the auto-correlation matrix R=x\*xT.

As can see in the equation that the iteration of weights only has a matrix multiplication. So the asymptotic complexity of the algorithm is O(N), i.e., linear in the number of weights. [1]

## Normalized LMS-NLMS

The NLMS is an extension of the LMS algorithm which bypasses this issue by calculating maximum step size value. [2] Step size value is calculated by using the following formula.

Step size=1/dot product (input vector, input vector) (2.3)

This step size is proportional to the inverse of the total expected energy of the input vector x(n). The sum of these expected energies is also equal to the trace of auto-correlation matrix R of input vector, which is

 (2.4)

The recursion formula for the NLMS algorithm is

w(n+1)=w(n)+2\*μ/x’(n)x(n)\*e(n)\*x(n) (2.5)

in which,

e(n)=d(n)-w(n)’\*x(n) (2.6)

## Cross-validation

The goal of cross-validation is to estimate how good a model fit a new data set, by new I mean independent of the training data set. The basic idea is to separate the original data set into several subsets, then pick one subset out as the validation data set or test data set. Use the left dataset as training set to get parameters. Apply these parameters into the validation set to get a estimate function (in this project, we calculate the ERLE). Repeat the process above for multiple rounds, the validation result is the average of all results over rounds. One problem that cross-validation can solve is overfitting. [3]

## Overfitting

When overfitting happens, the model is going to remember exactly what the training data are and thus performing very well on training data, however it will match the new data badly. The overfitting model is not able to generalize from trend.

# *Implement of the algorithm*

## Performance Surface for filter oder of 2

Firstly, μ has to be calculated by the following formula,

μ=μ0/tr(R) (3.1)

Practically μ0 is chosen from 0.01 to 0.5, in this case, four different values are set for μ0=[0.01,0.1, 0.3, 0.5]. Thus, four different values for μ are given. μ=[0.0047, 0.0467, 0.1401,0.2335]. By plotting their learning curve, as shown in Figure2, the best μ is 0.0047. As we can see in Figure2, The MSE for the best learning rate 0.0047 converged in about 2000 iterations and for other learning rate, the filter didn’t converge. The reason for the sudden increase of MSE is that the learning rate is set too large that the model can’t handle with the sudden change of the trend of the signal, in this case, the speech signal. When an irrelevant signal, the speech signal, is added to the noise, it will appear a big error in iteration. If the step size is big, it will cause the learning curve increasing suddenly.

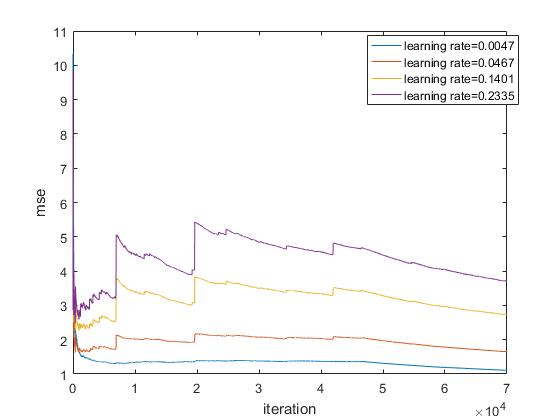


Figure 2 Learning curve for different learning rate

Setting μ=0.0047, plotting weight tracks is now available using formula(2.5). Figure 3 and 4 give the values of w1 and w2 over iterations.

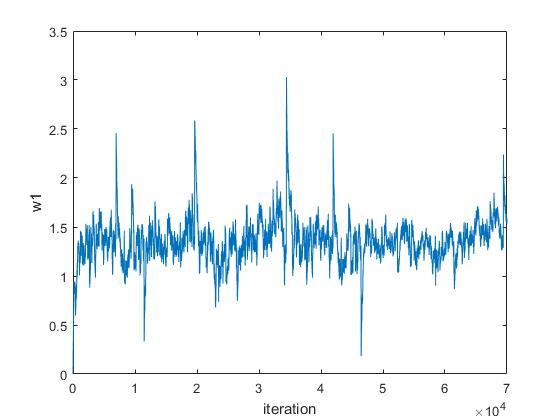


Figure 3 W1 over iteration

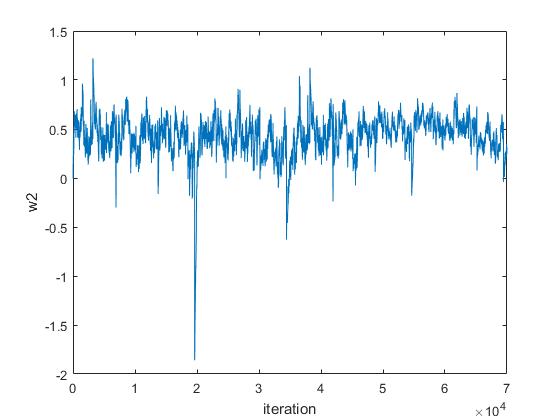


Figure 4 W2 over iteration

We can now see that w1 is ranging from 0.5 to 2 and w2 is ranging from -1 to 1. Then choose 100 different sets of w1 and w2, calculating the mean square error so that the performance surface can be plotted.

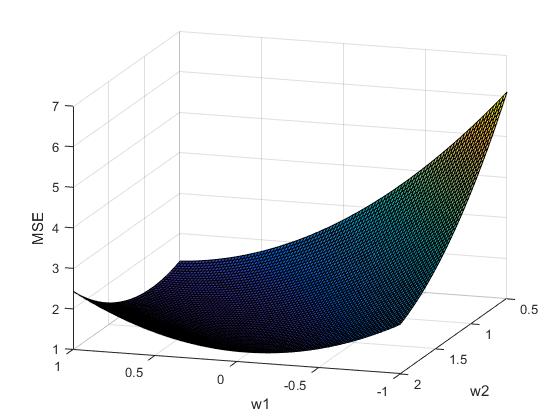


Figure 5 Performance Surface for filter order of 2.

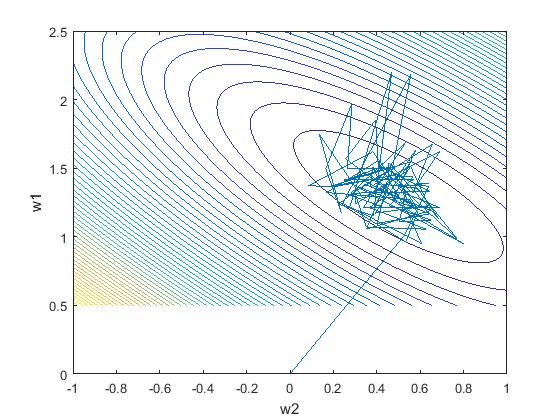


Figure 6 The contour and weights track

The performance shows that the MSE is quadratic of weights w and the w is converging to the optimal w\*.

## Estimate SNR by ERLE

The formula for Echo Return Loss Enhancement(ERLE) is,

 (3.2)

in which E[d^2] is the expected value for the signal power of primary input and E[e^2] is the expected value for the power of output, in this case E[e^2]=MSE.

For the unprocessed signal, the original ERLE is,

OERLE=10\*log(reference^2/(mean(reference-primary).^2)

=3.1421 dB

For the filter order of 2 and learning rate of 0.0047, the ERLE is 5.100dB. With this ERLE improvement, the sentence which the person said couldn’t be told.

## Larger Filter Order

|  |  |  |  |
| --- | --- | --- | --- |
| Filter Order | 10 | 20 | 50 |
| ERLE(dB) | 18.76 | 18.80 | 18.58 |

The ERLEs above are calculated with the same step-size 0.1. With these values of ERLE, it can be told that there’s someone talking, however the content is not clear enough.

This time, we fixed the filter order and adjust the step size. For each filter order, step-size is ranging from 0.1 to 2 with sample step size 0.1. Figure 7,8,9 show the ERLE plots respectively.

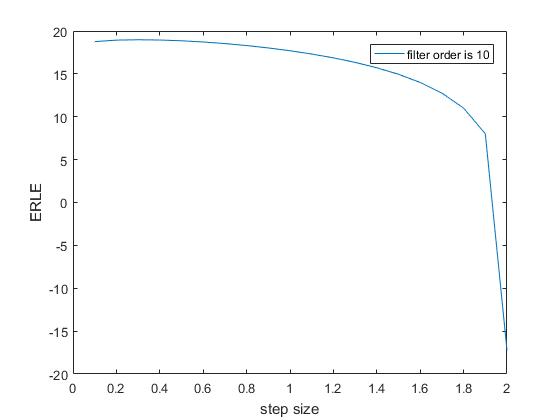


Figure 7

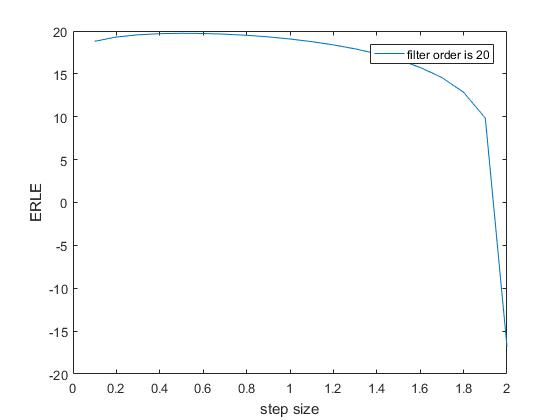


Figure 8

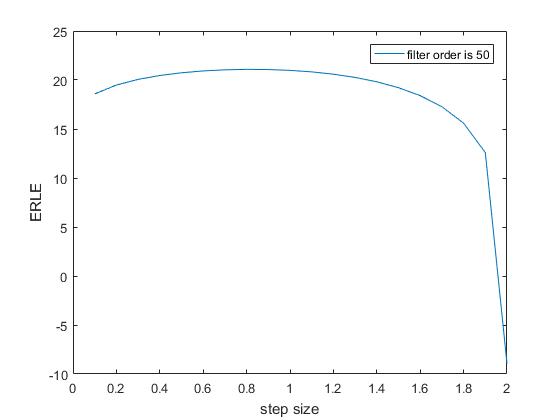


Figure 9

As we can see in these figures, filter order 50 has the biggest ERLE, which is 21.07 when the learning rate is 0.9. Thus, we’re going to focus on the situation when filter order is 50. To validate that, we still need to plot the learning curve. Figure 10 shows the effects of the different learning rates on the learning curve.

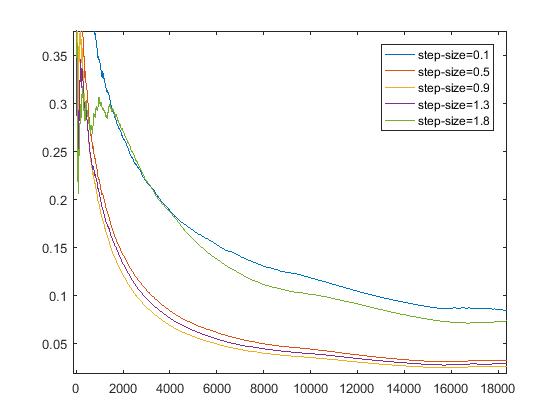


Figure 10

The step-size 0.9 does perform better in MSE converging.

# Result

## A.Signals in Time and Frequency Domain

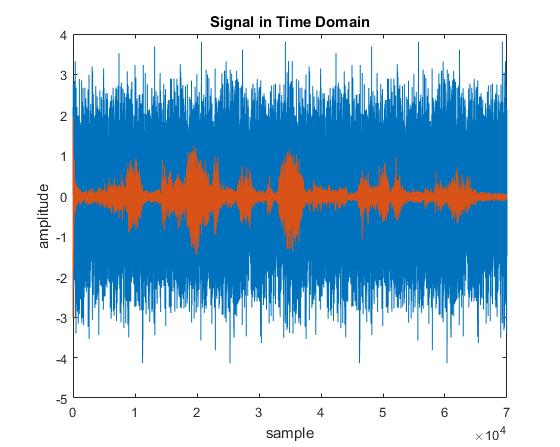
Figure 11 shows the reference signal ( speech + noise) and the filtered speech signal (‘error’) in time domain. 

Figure 11 orange signal is the filtered signal, blue is for original reference signal.

Figure 12 shows the reference signal (speech + noise) and the filtered speech signal (‘error’) in frequency domain.

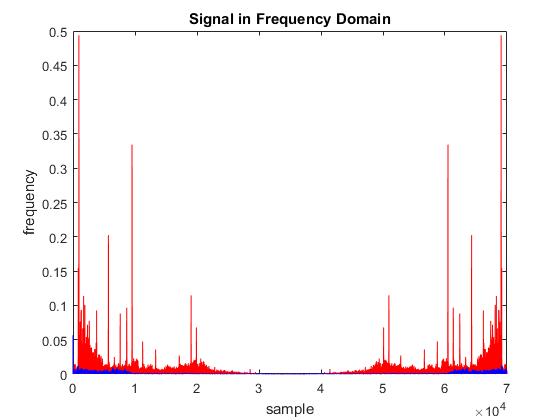


Figure 12 blue signal is the filtered signal, red is for original reference signal.

Figure 11 and 12 are plotted when filter order is 50 and step size is 0.9. In this condition, the content of the speech can be told. The ERLE for this is 21.07.

Here comes another set of filter order and step-size. When filter order is 50 and step-size is 0.1, Figure 13 gives the time domain signal comparison between primary signal and filtered signal while Figure14 gives the frequency domain comparison. The ERLE for this condition is 18.58.

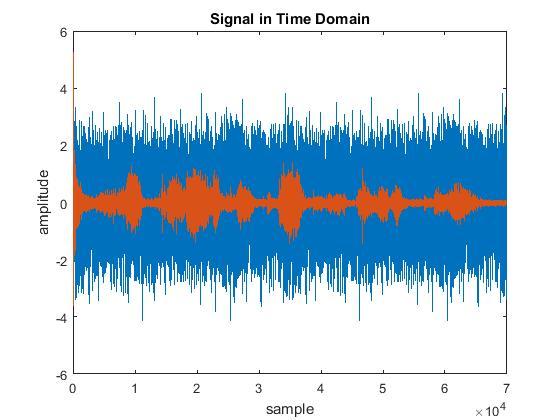


Figure 13 orange signal is the filtered signal, blue is for original reference signal.

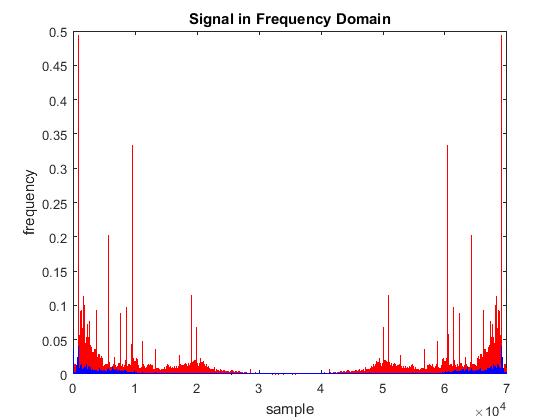


Figure 14 blue signal is the filtered signal, red is for original reference signal.

Although the ERLE of step-size 0.9 is larger than ERLE of step-size 0.1, the speech intelligibility when step-size is 0.9 is worse than when step- size is 0.1. By plotting both filtered signal with step-size 0.9 and 0.1 in the same figure, we can tell the slightly difference between them. Figure 15 shows these signals in time domain.

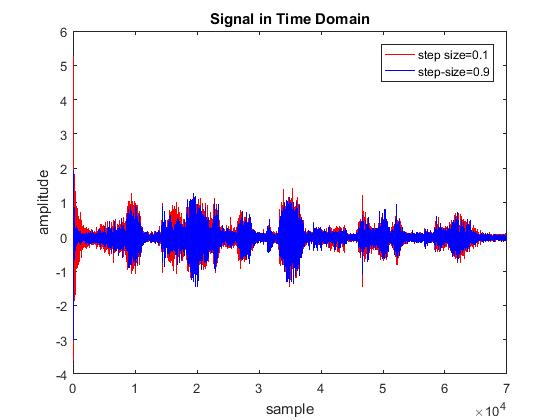


Figure 15 filtered signals from various step sizes.

# conclusion

The NLMS algorithm shows a good performance in interference canceling adaptive filter. The converging of MSE means the uncorrelated noise is filtering out from the reference signal(speech+noise). We can use the ERLE improvement to approximately estimate the performance of interference canceling adaptive filter. The ERLE enhanced by about 17dB after the NLMS adaptive filter. In this case, the content of speech can be easily told. It’s **“ I will not condone a course of action that will lead us to war**.”

The higher ERLE means lower MSE, which means less interference. This can be easily observed when we look into the formula for ERLE.

We can use the ERLE to cross validate the data set. By compare the original ERLE with the average ERLE from multiple rounds of cross-validation, it can show whether the model is overfitting. Here shows a simple 2-fold cross-validation for step size 0.9 and 0.1.

|  |  |  |
| --- | --- | --- |
| Step-size | 0.1 | 0.9 |
| Original ERLE | 18.58 | 21.07 |
| Average ERLE | 19.71 | 21.59 |

The inconsistency between ERLE and the intelligibility of filtered signal may cause by subjective judgement. The better signal for computer may not agree with the better signal for us. Also we can see from figure 14, the filter with step-size 0.9 has less noise than that with step-size 0.1, however with lower noise, filter with step-size 0.9 may filter out some information in speech signal which may reduce the intelligibility of speech. The difference in level of filtering quality is obvious when looking at the beginning of Figure14. The converge rate of filter with step-size 0.9 is faster, that may explain for the higher value of ERLE.

# Reference

[1] Jose. C. Principe , “lms algorithm.pdf”, spring 90.

[2]Gopalaiah, Dr.K. Suresh, “real time imlementation of adaptive filter algorithms for acoustic echo cancellation in telecommunication systems”, [J], May 2014, International Journal of Combined Research & Development (IJCRD ),Volume: 2; Issue: 5 .

[3][https://en.wikipedia.org/wiki/Cross-validation\_(statistics) - Measures\_of\_fit](https://en.wikipedia.org/wiki/Cross-validation_(statistics)#Measures_of_fit)