Acoustic Echo Cancellation Using Conventional Adaptive Algorithms

Raghava Yathiraju

Abstract - An adaptive filter is a filter that self-adjusts its transfer function according to an optimization algorithm driven by an error signal.Because of the complexity of the optimization algorithms,most adaptive filters are digital filters.Adaptive filtering constitutes one of the core technologies in digital signal processing and finds numerous application areas in science as well as in industry. Adaptive filtering techniques are used in a wide range of applications, including , adaptive noise cancellation, echo cancellation, adaptive equalization and adaptive beamforming. Acoustic echo cancellation is a common occurrence in today's telecommunication systems. The signal interference caused by acoustic echo is distracting to users and causes a reduction in the quality of the communication. This paper focuses on the use of Least Mean Square (LMS), Normalised Least Mean Square (NLMS), Variable Step-Size Least Mean Square (VSLMS), Variable Step-Size Normalised Least Mean Square (VSNLMS) and Recursive Least Square (RLS) algorithms to reduce this unwanted echo, thus increasing communication quality.

Keywords: Adaptive filters, Echo, Adaptive algorithms, Echo cancellation, Acoustic echo cancellation.

I. INTRODUCTION

Acoustic echo occurs when an audio signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time delayed images of the signal [1]. This paper will focus on the occurrence of acoustic echo in telecommunication systems. Adaptive filters are dynamic filters which iteratively alter their characteristics in order to achieve an optimal desired Output. An adaptive filter algorithmically alters its parameters in order to minimise a function of the difference between the desired output d (n) and its actual output y (n).

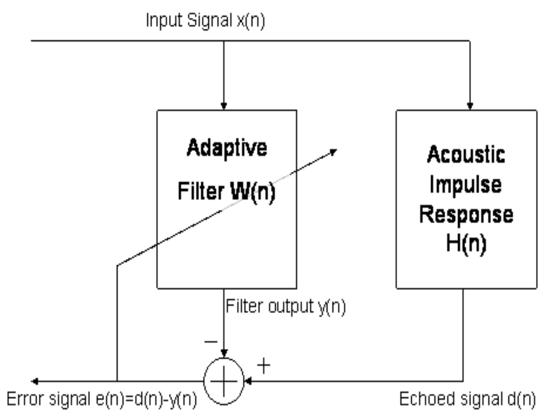

Fig. 1. Adaptive echo Cancellation system

Manuscript received November, 2013.
Raghava Yathiraju, Vice Principal, St.Mary's Group of Institutions. Guntur, India.

This function is known as the cost function of the adaptive algorithm. Fig. 1 shows a block diagram of the adaptive echo cancellation system. Here the filter H (n) represents the impulse response of the acoustic environment, W(n) represents the adaptive filter used to cancel the echo signal. The adaptive filter aims to equate its output y(n) to the desired output d(n) (the signal reverberated within the acoustic environment). At each iteration the error signal, e (n) =d (n)-y (n), is fed back into the filter, where the filter characteristics are altered accordingly [2-3].

The aim of an adaptive filter is to calculate the difference between the desired signal and the adaptive filter output, e(n). This error signal is fed back into the adaptive filter and its coefficients are changed algorithmically in order to minimise a function of this difference, known as the cost function. In the case of acoustic echo cancellation, the optimal output of the adaptive filter is equal in value to the unwanted echoed signal. When the adaptive filter output is equal to desired signal the error signal goes to zero. In this situation the echoed signal would be completely cancelled and the far user would not hear any of their original speech returned to them.

II. LEAST MEAN SQUARE (LMS) ALGORITHM

The Least Mean Square (LMS) algorithm was first developed by Widrow and Hoff in 1959 through their studies of pattern recognition. From there it has become one of the most widely used algorithms in adaptive filtering. The LMS algorithm is a type of adaptive filter known as stochastic gradient-based algorithms as it utilises the gradient vector of the filter tap weights to converge on the optimal wiener solution. It is well known and widely used due to its computational simplicity. It is this simplicity that has made it the benchmark against which all other adaptive filtering algorithms are judged [4]. With each iteration of the LMS algorithm, the filter tap weights of the adaptive filter are updated according to the following formula.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \quad (1)$$

Here x(n) is the input vector of time delayed input values, $x(n) = [x(n)\ x(n-1)\ x(n-2)\ ..\ x(n-N+1)]^T$. The vector $w(n) = [w_0(n)\ w_1(n)\ w_2(n)\ ..\ w_{N-1}(n)]^T$ represents the coefficients of the adaptive FIR filter tap weight vector at time n. The parameter μ is known as the step size parameter and is a small positive constant. This step size parameter controls the influence of the updating factor. Selection of a suitable value for μ is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if μ is too large the adaptive filter becomes unstable and its output diverges.

2.1. *Implementation of the LMS algorithm*

Each iteration of the LMS algorithm requires 3 distinct steps in this order:
1. The output of the FIR filter, y(n) is calculated using eq. (2).

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{2}$$

2. The value of the error estimation is calculated using eq. (3).

$$e(n)=d(n)-y(n) \tag{3}$$

3. The tap weights of the FIR vector are updated in preparation for the next iteration, by eq. (4).

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \tag{4}$$

The main reason for the LMS algorithms popularity in adaptive filtering is its computational simplicity, making it easier to implement than all other commonly used adaptive algorithms. For each iteration the LMS algorithm requires 2N additions and 2N+1 multiplications (N for calculating the output, y(n), one for 2μe(n) and an additional N for the scalar by vector multiplication [5-7].

III. NORMALISED LEAST MEAN SQUARE (NLMS) ALGORITHM

One of the primary disadvantages of the LMS algorithm is having a fixed step size parameter for every iteration. This requires an understanding of the statistics of the input signal prior to commencing the adaptive filtering operation. In practice this is rarely achievable [8-9]. Even if we assume the only signal to be input to the adaptive echo cancellation system is speech, there are still many factors such as signal input power and amplitude which will affect its performance.
The normalised least mean square algorithm (NLMS) is an extension of the LMS algorithm which bypasses this issue by calculating maximum step size value. Step size value is calculated by using eq. (9). This step size is proportional to the inverse of the total expected energy of the instantaneous values of the coefficients of the input vector x(n). This sum of the expected energies of the input samples is also equivalent to the dot product of the input vector with itself, and the trace of input vectors auto-correlation matrix, R.

$$\begin{aligned} tr[\mathbf{R}] &= \sum_{i=0}^{N-1} E\left[x^2(n-i)\right] \\ &= E\left[\sum_{i=0}^{N-1} x^2(n-i)\right] \end{aligned} \tag{5}$$

The recursion formula for the NLMS algorithm is stated in eq. (6).

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} e(n)\mathbf{x}(n) \tag{6}$$

3.1. Implementation of the NLMS algorithm

The NLMS algorithm has been implemented in Matlab. As the step size parameter is chosen based on the current input values, the NLMS algorithm shows far greater stability with unknown signals. This combined with good convergence speed and relative computational simplicity makes the NLMS algorithm ideal for the real time adaptive echo cancellation system [10].
As the NLMS is an extension of the standard LMS algorithm, the NLMS algorithms practical implementation is very similar to that of the LMS algorithm. Each iteration of the NLMS algorithm requires these steps in the following order.
1. The output of the adaptive filter is calculated.

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{7}$$

2. An error signal is calculated as the difference between the desired signal and the filter output.

$$e(n) = d(n) - y(n) \tag{8}$$

3. The step size value for the input vector is calculated.

$$\mu(n) = \frac{1}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{x}(n)} \tag{9}$$

4. The filter tap weights are updated in preparation for the next iteration.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n)e(n)\mathbf{x}(n) \tag{10}$$

Each iteration of the NLMS algorithm requires 3N+1 multiplications, this is only N more than the standard LMS algorithm. This is an acceptable increase considering the gains in stability and echo attenuation achieved.

IV. VARIABLE STEP SIZE LMS (VSLMS) ALGORITHM

Both the LMS and the NLMS algorithms have a fixed step size value for every tap weight in each iteration. In the Variable Step Size Least Mean Square (VSLMS) algorithm the step size for each iteration is expressed as a vector, μ(n). Each element of the vector μ(n) is a different step size value corresponding to an element of the filter tap weight vector, w(n) [11-12].

4.1. *Implementation of the VSLMS algorithm*

The VSLMS algorithm is executed by following these steps for each iteration. With ρ=1, each iteration of the VSLMS algorithm requires 4N+1 multiplication operations [13-14].
1. The output of the adaptive filter is calculated.

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{11}$$

2. The error signal is calculated as the difference between the desired output and the filter output.

$$e(n) = d(n) - y(n) \tag{12}$$

3. The gradient, step size and filter tap weight vectors are updated using the following equations in preparation for the next iteration.

$$\begin{aligned}
&\text{For } i = 0,1,\ldots,N-1\\
&g_i(n) = e(n)x(n-i)\\
&\mathbf{g}(n) = e(n)\mathbf{x}(n)\\
&\mu_i(n) = \mu_i(n-1) + \rho g_i(n) g_i(n-1)\\
&\text{if } \mu_i(n) > \mu_{\max}, \mu_i(n) = \mu_{\max}\\
&\text{if } \mu_i(n) < \mu_{\min}, \mu_i(n) = \mu_{\min}\\
&w_i(n+1) = w_i(n) + 2\mu_i(n) g_i(n)
\end{aligned} \tag{13}$$

V. VARIABLE STEP SIZE NORMALISED LMS (VSNLMS) ALGORITHM

The VSLMS algorithm still has the same drawback as the standard LMS algorithm in that to guarantee stability of the algorithm, a statistical knowledge of the input signal is required prior to the algorithms commencement. Also, recall the major benefit of the NLMS algorithm is that it is designed to avoid this requirement by calculating an appropriate step size based upon the instantaneous energy of the input signal vector.

It is a natural progression to incorporate this step size calculation into the variable step size algorithm, in order increase stability for the filter without prior knowledge of the input signal statistics. This is what I have tried to achieve in developing the Variable step size normalised least mean square (VSNLMS) algorithm [15-16].

In the VSNLMS algorithm the upper bound available to each element of the step size vector, μ(n), is calculated for each iteration. As with the NLMS algorithm the step size value is inversely proportional to the instantaneous input signal energy.

5.1. Implementation of the VSNLMS algorithm

The VSNLMS algorithm is implemented in Matlab. It is essentially an extension of the implementation of the VSLMS algorithm with the added calculation of a maximum step size parameter for each iteration [17-18].

1. The output of the adaptive filter is calculated.

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{14}$$

2. The error signal is calculated as the difference between the desired output and the filter output.

$$e(n) = d(n) - y(n) \tag{15}$$

3. The gradient, step size and filter tap weight vectors are updated using the following equations in preparation for the next iteration.

$$\begin{aligned}
&\text{for } i = 0,1,\ldots,N-1\\
&g_i(n) = e(n)x(n-i)\\
&\mathbf{g}(n) = e(n)\mathbf{x}(n)\\
&\mu_i(n) = \mu_i(n-1) + \rho g_i(n) g_i(n-1)\\
&\mu_{\max}(n) = \frac{1}{2\mathbf{x}^T(n)\mathbf{x}(n)}\\
&\text{if } \mu_i(n) > \mu_{\max}(n), \mu_i(n) = \mu_{\max}(n)\\
&\text{if } \mu_i(n) < \mu_{\min}(n), \mu_i(n) = \mu_{\min}(n)\\
&w_i(n+1) = w_i(n) + 2\mu_i(n) g_i(n)
\end{aligned} \tag{16}$$

ρ is an optional constant the same as is the VSLMS algorithm. With ρ =1, each iteration of the VSNLMS algorithm requires 5N+1 multiplication operations.

VI. RECURSIVE LEAST SQUARE (RLS) ALGORITHM

These algorithms attempt to minimise the cost function in eq. (17), where k=1 is the time at which the RLS algorithm commences and λ is a small positive constant very close to, but smaller than 1. With values of λ<1 more importance is given to the most recent error estimates and thus the more recent input samples, this results in a scheme that places more emphasis on recent samples of observed data and tends to forget the past [19].

$$\zeta(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^2(k) \tag{17}$$

Unlike the LMS algorithm and its derivatives, the RLS algorithm directly considers the values of previous error estimations. RLS algorithms are known for excellent performance when working in time varying environments. These advantages come with the cost of an increased computational complexity and some stability problems.

6.1. *Implementation of the RLS algorithm*

As stated the previously the memory of the RLS algorithm is confined to a finite number of values, corresponding to the order of the filter tap weight vector. Firstly, two factors of the RLS implementation should be noted: the first is that although matrix inversion is essential to the derivation of the RLS algorithm, no matrix inversion calculations are required for the implementation, thus greatly reducing the amount of computational complexity of the algorithm [20]. Secondly, unlike the LMS based algorithms, current variables are updated within the iteration they are to be used, using values from the previous iteration.

To implement the RLS algorithm, the following steps are executed in the following order.

1. The filter output is calculated using the filter tap weights from the previous iteration and the current input vector.

$$\overline{y}_{n-1}(n) = \overline{\mathbf{w}}^T(n-1)\mathbf{x}(\mathbf{n}) \tag{18}$$

2. The intermediate gain vector is calculated using eq. (19).

$$\mathbf{u}(n) = \tilde{\psi}_\lambda^{-1}(n-1)\mathbf{x}(n)$$

$$\mathbf{k}(n) = \frac{1}{\lambda + \mathbf{x}^T(n)\mathbf{u}(n)}\mathbf{u}(n) \tag{19}$$

3. The estimation error value is calculated using eq. (20).

$$\overline{e}_{n-1}(n) = d(n) - \overline{y}_{n-1}(n) \tag{20}$$

4. The filter tap weight vector is updated using eq. (21) and the gain vector is calculated in eq. (19).

$$\mathbf{w}(n) = \overline{\mathbf{w}}^T(n-1) + \mathbf{k}(n)\overline{e}_{n-1}(n) \tag{21}$$

5. The inverse matrix is calculated using eq. (22).

$$\psi_\lambda^{-1}(n) = \lambda^{-1}(\psi_\lambda^{-1}(n-1) - \mathbf{k}(n)[\mathbf{x}^T(n)\psi_\lambda^{-1}(n-1)])$$

Each iteration of the RLS algorithm requires $4N^2$ operations and $3N^2$ additions. This makes its very costly to implement, thus LMS based algorithms,

VII. RESULTS OF LMS ALGORITHM

The LMS algorithm was simulated using Matlab. Fig. 2 shows the input speech signal which is collected from the computer system through microphone and is common to all these algorithms. Fig. 3 shows the desired echo signal derived from the input signal and is common to all these algorithms. Fig. 4 shows the adaptive filter output which will reduce the echo signal from the input signal. Fig. 5 shows the mean square error signal calculated from the filter output signal. Fig. 6 shows the attenuation which is derived from the division of echo signal to the error signal. The adaptive filter is a 1000th order FIR filter. The step size was set to 0.05. The MSE shows that as the algorithm progresses the average value of the cost function decreases.

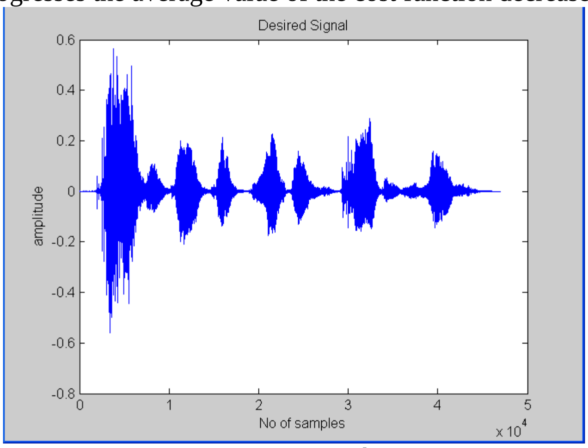

Fig.2.InputSignal

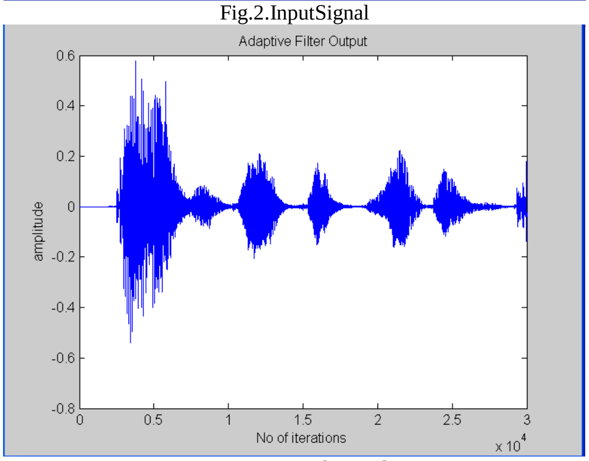

Fig. 3. Desired Signal

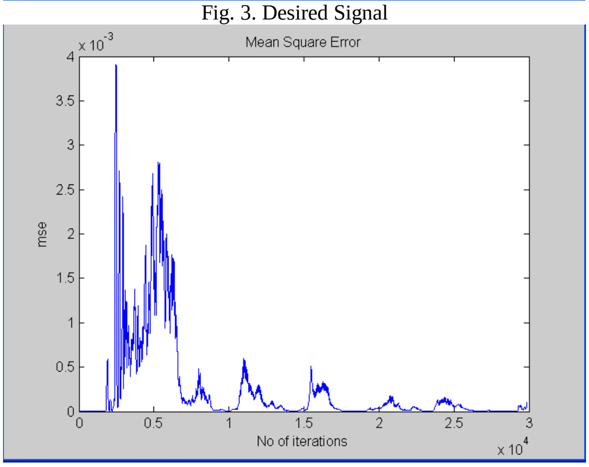

Fig. 4. Adaptive Filter Output

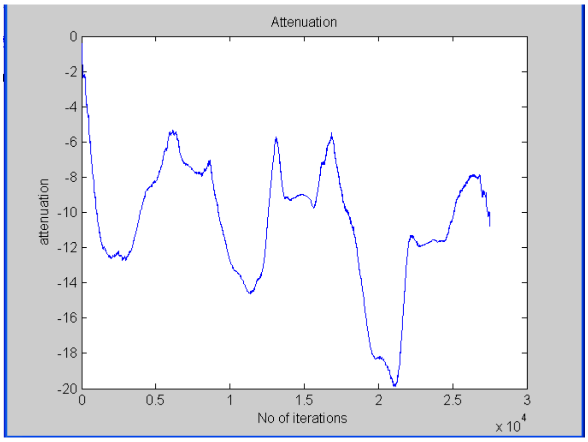

Fig. 5. Mean Square Error

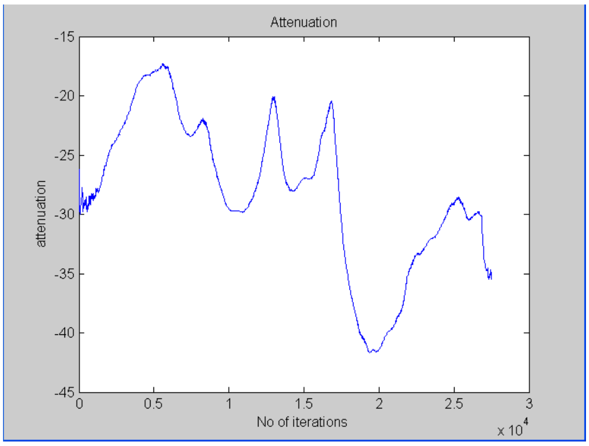

Fig. 6. Attenuation

The NLMS algorithm was simulated using Matlab. Fig. 7 shows the adaptive filter output. Fig. 8 shows the mean square error. Fig. 9 shows the attenuation. The adaptive filter is a 1000th order FIR filter.

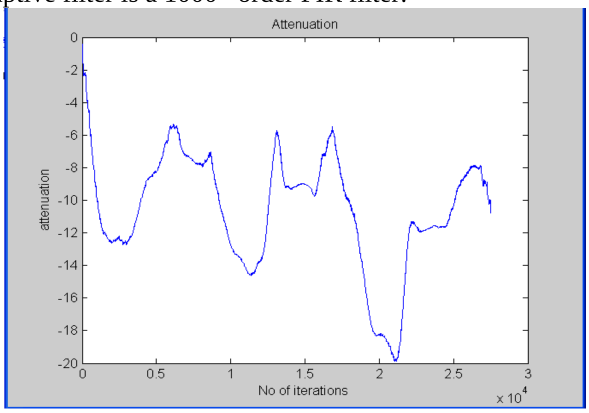

Fig. 7. Adaptive Filter Output

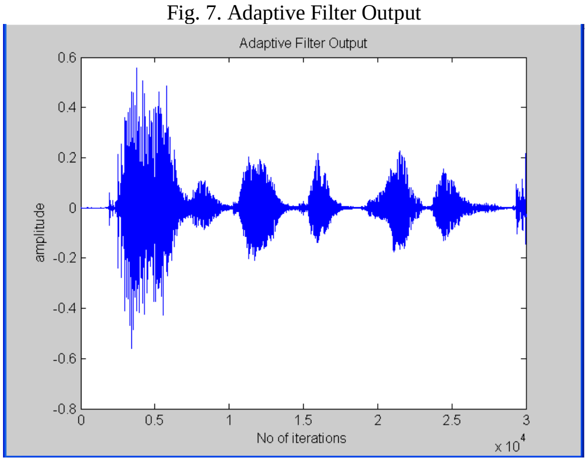

Fig. 7. Adaptive Filter Output

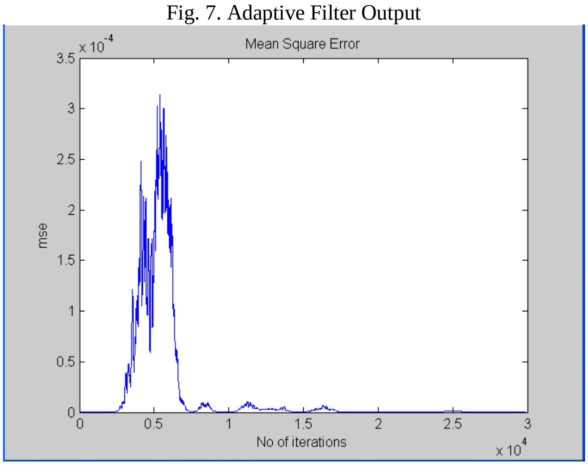

Fig. 8. Mean Square Error

Fig. 9. Attenuation

Fig. 8. Mean Square Error

XI. RESULTS OF VSLMS ALGORITHM

The VSLMS algorithm was simulated using Matlab. Fig. 10 shows the adaptive filter output which will reduce the echo signal from the input signal. Fig. 11 shows the mean square error signal calculated from the filter output signal. Fig. 12 shows the attenuation which is derived from the division of echo signal to the error signal. The adaptive filter is a 1000th order FIR filter. The upper step size was set to 0.1 and the lower step
size was set to 0.0001.

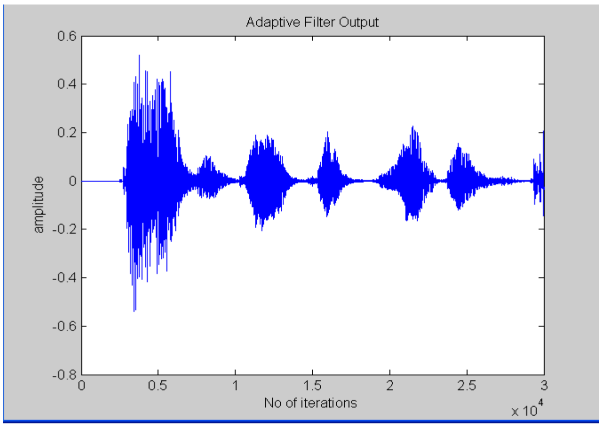

Fig. 10. Adaptive Filter Output

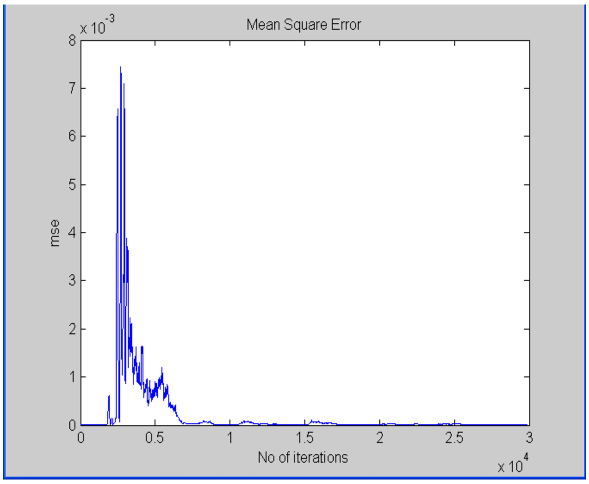

Fig. 11. Mean Square Error

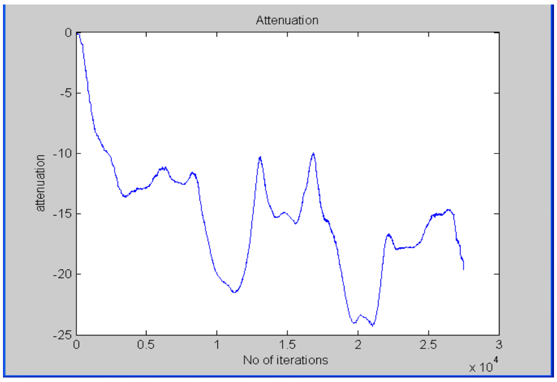

Fig. 12. Attenuation

X. RESULTS OF VSNLMS ALGORITHM

The VSNLMS algorithm was simulated using Matlab. Fig. 13 shows the adaptive filter output which will reduce the echo signal from the input signal. Fig. 14 shows the mean square error signal calculated from the filter output signal. Fig. 15 shows the attenuation which is derived from the division of echo signal to the error signal. The adaptive filter is a 1000th order FIR filter. The lower step size was set to 0.0001 and the upper step size value is calculated using eq. (16).

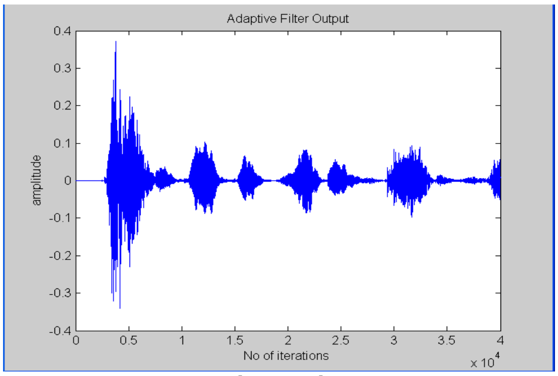

Fig. 13. Adaptive Filter Output

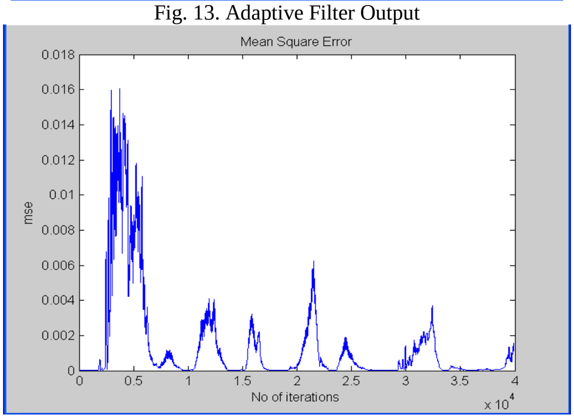

Fig. 14. Mean Square Error

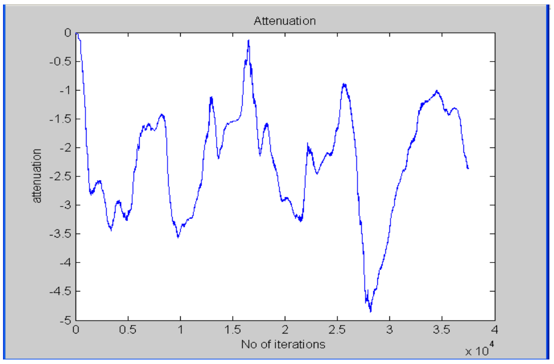

Fig. 15. Attenuation

XI. RESULTS OF RLS ALGORITHM

The RLS algorithm was simulated using Matlab. Fig. 16 shows the adaptive filter output which will reduce the echo signal from the input signal. Fig. 17 shows the mean square error signal calculated from the filter output signal. Fig. 18 shows the attenuation which is derived from the division of echo signal to the error signal. The adaptive filter is a 1000^{th} order FIR filter. The lambda value was set to 0.99.

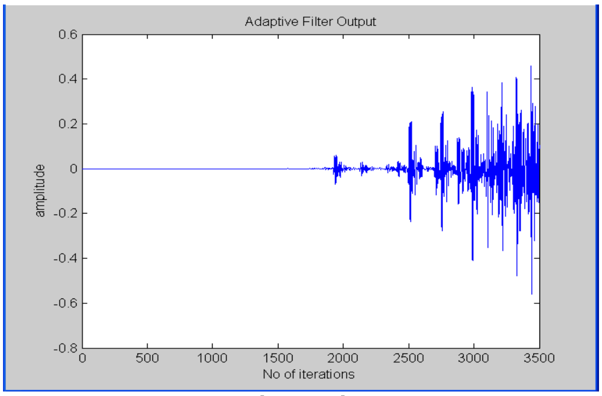

Fig. 16. Adaptive Filter Output

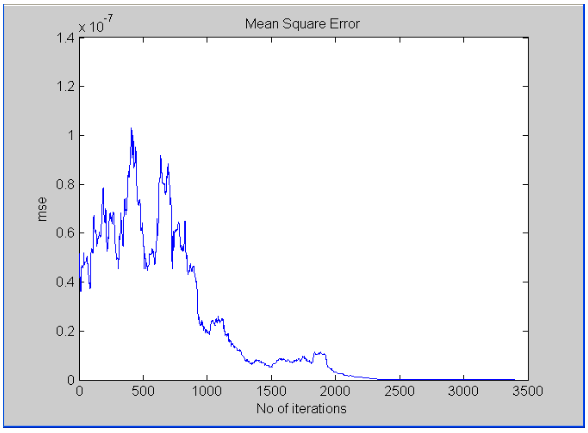

Fig 17 Mean Square error.

Table 1. Summary of adaptive algorithms performance

ALGORITHM	ITERATIONS	FILTER ORDER	MEAN SQUARE ERROR	AVERAGE ATTENUATION	COMPUTATIONS
LMS	30000	1000	0.004	-12dB	2N+1
NLMS	30000	1000	0.0003	-28dB	3N+1
VSLMS	30000	1000	0.0075	-10dB	4N+1
VSNLMS	40000	1000	0.016	-3dB	5N+1
RLS	3500	1000	0.0000001	-40dB	$4N^2$

XII. CONCLUSIONS

Because of its simplicity, the LMS algorithm is the most popular adaptive algorithm. However, the LMS algorithm suffers from slow and data dependent convergence behavior.
The NLMS algorithm, an equally simple, but more robust variant of the LMS algorithm, exhibits a better balance between simplicity and performance than the LMS algorithm. Due to its good properties the NLMS has been largely used in real-time applications.
Because the step size value is variable it doesn't require understanding of the statistics of the input signal prior to commencing the adaptive filter operation. This is very poor considering each iteration of the VSLMS algorithm has 2N more multiplication operations than the LMS algorithm. This is possibly due to speech signals being non-stationary, regardless the VSLMS aglorithm was not considered for the real time application.
VSNLMS algorithm has poor performance when compared to all these algorithms. VSNLMS algorithm requires more number of multiplications than LMS, NLMS and VSLMS algorithms.
The RLS algorithm has the greatest attenuation of all algorithms and converges much faster than LMS, NLMS and RLS algorithms. Due to the large number of multiplications it is rather costly to be implemented.

REFERENCES

[1] Homana, I.; Topa, M.D.; Kirei, B.S.; "Echo cancelling using adaptive algorithms", Design and Technology of Electronics Packages, (SIITME) 15th International Symposium., pp. 317-321, Sept.2009.
[2] Eneman, K.; Moonen, M.; "Iterated partitioned block frequency-domain adaptive filtering for acoustic echo cancellation," IEEE Transactions on Speech and Audio Processing, vol. 11, pp. 143-158, March 2003.
[3] G. Egelmeers, P. Sommen, and J. de Boer, "Realization of an acoustic echo canceller on a single DSP," in Proc. Eur. Signal Processing Conf. (EUSIPCO96), Trieste, Italy, pp. 33–36, Sept. 1996.
[4] Soria, E.; Calpe, J.; Chambers, J.; Martinez, M.; Camps, G.; Guerrero, J.D.M.; "A novel approach to introducing adaptive filters based on the LMS algorithm and its variants", IEEE Transactions, vol. 47, pp. 127-133, Feb 2008.
[5] Krishna, E.H.; Raghuram, M.; Madhav, K.V; Reddy, K.A; "Acoustic echo cancellation using a computationally efficient transform domain LMS adaptive filter," 2010 10th International Conference on Information sciences signal processing and their applications (ISSPA), pp. 409-412, May 2010.
[6] E. Soria, J. Calpe, J. Guerrero, M. Martínez, and J. Espí, "An easy demonstration of the optimum value of the adaptation constant in the LMS algorithm," IEEE Trans. Educ., vol. 41, pp. 83, Feb. 1998.
[7] D. Morgan and S. Kratzer, "On a class of computationally efficient rapidly converging, generalized NLMS algorithms," IEEE Signal Processing Lett., vol. 3, pp. 245–247, Aug. 1996
[8] Tandon, A.; Ahmad, M.O.; Swamy, M.N.S.; "An efficient, low-complexity, normalized LMS algorithm for echo cancellation", IEEE workshop on Circuits and Systems, 2004. NEWCAS 2004, pp. 161-164, June 2004.
[9] Lee, K.A.; Gan, W.S; "Improving convergence of the NLMS algorithm using constrained subband updates," Signal Processing Letters IEEE, vol. 11, pp. 736-739, Sept. 2004.
[10] D.L. Duttweiler, "Proportionate Normalized Least Mean Square Adaptation in Echo Cancellers," IEEE Trans. Speech Audio Processing, vol. 8, pp. 508-518, Sept. 2000.
[11] Sristi, P.; Lu, W.-S.; Antoniou, A.;"A new variable step-size LMS algorithm and its application in subband adaptive filtering for echo cancellation," The 2001 IEEE International Synposium on Circuits and Systems, 2001. ISCAS 2001, vol. 2, pp. 721-724, May 2001.
[12] Tingchan, W.; Chutchavong, V.; Benjangkaprasert, C.; " Performance of A Robust Variable Step-Step LMS Adaptive Algorithm for multiple Echo Cancellation in Telephone Network," SICE-ICASE, 2006. International Joint Conference, pp. 3173-3176, Oct 2006.
[13] Li Yan; Wang Xinan; "A Modified VSLMS Algorithm," The 9th International Conference on Advanced Communication Technology, vol. 1, pp. 615-618, Feb 2007.

[14] J. B. Evans, P. Xue, and B. Liu, “Analysis and implementation of variable step size adaptive algorithms,” IEEE Trans. Signal Processing, vol. 41, pp. 2517– 2535, Aug. 1993.
[15] Paleologu, C.; Benesty, J.; Grant, S.L.; Osterwise, C.; “Variable step-size NLMS algorithms for echo cancellation” 2009 Conference Record of the forty-third Asilomar Conference on Signals, Systems and Computers., pp. 633-637, Nov 2009.
[16] Ahmed I. Sulyman and Azzedine Zerguine, "Echo Cancellation Using a Variable Step-Size NLMS Algorithm", Electrical and Computer Engineering Department Queen's University.
[17] J. Benesty, H. Rey, L. Rey Vega, and S. Tressens, “A nonparametric VSS NLMS algorithm,” IEEE Signal Process. Lett., vol. 13, pp. 581–584, Oct. 2006.
[18] S.C. Douglas, “Adaptive Filters Employing Partial Updates,” IEEE Trans.Circuits SYS.II, vol. 44, pp. 209-216, Mar 1997.
[19] Jun Xu; Wei-ping Zhou; Yong Guo; “A Simplified RLS Algorithm and Its Application in Acoustic Echo Cancellation,” 2nd International conference on Information Engineering and Computer Science, pp. 1 4, Dec.2010.
[20] Mohammed, J.R.; Singh, G.; “An Efficient RLS Algorithm For Output-Error Adaptive IIR Filtering And Its Application To Acoustic Echo Cancellation,” IEEE Symposium on Computational Intelligence in Image and Signal Processing, 2007. CIISP 2007, pp. 139-145, April 2007.
[21] J. Shynk, “Frequency-domain and multirate adaptive filtering,” IEEE Signal Processing Mag., vol. 9, pp. 15– 37, Jan. 1992.
[22] D. L. Duttweiler, “A twelve-channel digital echo canceller,” IEEE Trans. Commun., vol. 26, no. 5, pp. 647–653, May 1978.