# Performance Study of LMS Based Adaptive Algorithms for Unknown System Identification

Shazia Javed and Noor Atinah Ahmad

*School of Mathematical Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia.*

**Abstract.** Adaptive filtering techniques have gained much popularity in the modeling of unknown system identification problem. These techniques can be classified as either iterative or direct. Iterative techniques include stochastic descent method and its improved versions in affine space. In this paper we present a comparative study of the least mean square (LMS) algorithm and some improved versions of LMS, more precisely the normalized LMS (NLMS), LMS-Newton, transform domain LMS (TDLMS) and affine projection algorithm (APA). The performance evaluation of these algorithms is carried out using adaptive system identification (ASI) model with random input signals, in which the unknown (measured) signal is assumed to be contaminated by output noise. Simulation results are recorded to compare the performance in terms of convergence speed, robustness, misalignment, and their sensitivity to the spectral properties of input signals. Main objective of this comparative study is to observe the effects of fast convergence rate of improved versions of LMS algorithms on their robustness and misalignment.

**Keywords:** Adaptive Filtering, misalignment, affine projections, system modeling.
**PACS:** 02.60.Ed

## INTRODUCTION

The stochastic gradient based least mean square (LMS) algorithm is the most famous adaptive algorithm in signal processing applications because of its easiest and reliable approach [1-5] . Although LMS algorithm is a robust and computationally simple algorithm, its performance is highly dependent on the spectral power (i.e. eigenvalue spread) of the input signal's autocorrelation matrix. The spectral power of input signals increases with an increase in correlation and this results in poor performance of LMS algorithm. Several iterative algorithms have been presented in literature to overcome this problem and have improved performance. Narayan proposed transform domain LMS (TD-LMS) algorithm using data-independent orthogonal transforms[6], while LMS-Newton algorithm is a good realization of the same algorithm[3]. These algorithms provided better convergence speed than that of conventional LMS algorithm. Further improvements in the convergence speed were made by Normalized LMS (NLMS) algorithm [7, 8] that updates the weight vector based on the current input vector, and Affine projection algorithm (APA) [9] that is based on affine subspace projections.

In this paper a comparative study of system identification of an unknown plant is performed, employing LMS algorithm and its four well known variants LMS-Newton, NLMS, TD-LMS and APA. The purpose of this study is to find out the characteristics of LMS algorithm which still need to be improved. Emphasis is given to the misalignment that is the tap-weight error of adaptive filter. In simulations, learning curves of normalized misalignment of all the algorithms are compared, and it is found that although improved versions of LMS algorithm have fast convergence speed for mean squares error (MSE), but none has exhibited better convergence in misalignment, than conventional LMS algorithm.

## LMS ADAPTIVE FILTER

Consider an FIR filter of length $N$ with a tap-weight vector $\mathbf{w}_n$, and error signal $e(n)=s(n)-\mathbf{w}_n^T\mathbf{a}_n$ at instant $n$. Here $s(n)$ is the output signal, while the input vector $\mathbf{a}_n=\left[\tilde{u}(n) \quad \tilde{u}(n-1) \quad \cdots \quad \tilde{u}(n-N+1)\right]^T$ is formed by the input signal $\tilde{u}(n)$. The LMS algorithm minimizes the instantaneous error function $J(n)=e^2(n)$, for

*Proceedings of the 21st National Symposium on Mathematical Sciences (SKSM21)*
AIP Conf. Proc. 1605, 233-238 (2014); doi: 10.1063/1.4887594

minimum MSE, and this minimization problem is equivalent to updating the tap-weight vector $\mathbf{w}_n$, as each new input signal is received, to get an optimal solution $\mathbf{w}_{opt}$. The update equation of the LMS algorithm is given as:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + 2\mu e(n)\mathbf{a}_n, \tag{1}$$

where $\mu$ is a positive constant that controls the rate of convergence. For stationary input and an appropriate choice of $\mu$, the minimum value of $e(n)$ generates a Cauchy sequence $\{\mathbf{w}_n\}_{n=1}^{\infty}$ from (1) in $\mathbb{R}^N$. But since $\mathbb{R}^N$ is a Banach space [10], there must exist an optimum weight vector $\mathbf{w}_{opt} \in \mathbb{R}^N$, such that $\mathbf{w}_n \to \mathbf{w}_{opt}$ as $n$ gets very large. Value of optimal solution $\mathbf{w}_{opt}$, as given by Wiener-Hopf equation [3], is $\mathbf{w}_{opt} = \mathbf{X}^{-1}\mathbf{p}$, where $\mathbf{X} = E\{\mathbf{a}_n\mathbf{a}_n^T\}$ is the autocorrelation matrix of input signal, and $\mathbf{p} = E\{s(n)\mathbf{a}_n\}$ is its crosscorrelation vector. Defining misalignment of LMS algorithm's tap weight vector as: $\mathbf{m}_n = \mathbf{w}_n - \mathbf{w}_{opt}$, and solving Eq- (1), [3] gives

$$E\{\mathbf{m}_{n+1}\} = (I - 2\mu\mathbf{X})E\{\mathbf{m}_n\}. \tag{2}$$

It is clear from Eq-(2) that $E\{\mathbf{m}_n\}$ forms a geometric progression, with common ratio equal to $\|I - 2\mu\mathbf{X}\|$. It implies that convergence behavior of LMS algorithm can be determined by the geometric progressions of filter weights $\mathbf{w}_n$ towards the optimal solution $\mathbf{w}_{opt}$. In that case $E\{\mathbf{m}_n\}$ converges to zero, and

$$|I - 2\mu\lambda_j| < 1 \quad \forall \quad j = 1, \ldots, N,$$

where $\lambda_j$ is an eigenvalue of the autocorrelation matrix $\mathbf{X} = E\{\mathbf{a}_n\mathbf{a}_n^T\}$. Simplification of above inequality gives $0 < \mu < \frac{1}{\lambda_j}$; $\forall\ j = 1, \ldots, N$. To ensure the stability of the adaptive process, value of $\mu$ must satisfy the condition:

$$0 < \mu < \frac{1}{\lambda_{\max}}; \tag{3}$$

$\lambda_{\max}$ is the largest eigenvalues of the autocorrelation matrix $\mathbf{X}$ and is given by the maximum of the power spectrum of input signal $\mathbf{a}_n$. Hence convergence rate of LMS algorithm depends upon the eigenvalue spread of the correlation matrix $\mathbf{X}$ of input signals. Since it is difficult to have a prior knowledge of initial settings of weight vector with respect to eigenvalue spread of matrix $\mathbf{X}$, it is difficult to predict the rate of convergence of LMS algorithm. This drawback has motivated researchers to look for some efficient modifications of computationally simple LMS algorithm, and several improvements have been developed so far. Here we present a few of them and see that still there is scope of further improvement in the efficiency of the algorithm.

## The LMS-Newton Algorithm

The input vector $\mathbf{a}_n$ in error term $\mu e(n)\mathbf{a}_n$ of Eq-(1), is preconditioned by an estimate of the inverse $\mathbf{X}_n^{-1}$ of input signal's autocorrelation matrix $\mathbf{X}$. The modified update equation is:

$$\mathbf{w}_{n+1}^{Nt} = \mathbf{w}_n^{Nt} + 2\mu e(n)\mathbf{X}_n^{-1}\mathbf{a}_n. \tag{4}$$

Misalignment $\mathbf{m}_n^{Nt} = \mathbf{w}_n^{Nt} - \mathbf{w}_{opt}$ of LMS-Newton algorithm, satisfies (by Eq-(2))

$$E\{\mathbf{m}_{n+1}^{Nt}\} = (I - 2\mu\mathbf{X}_n^{-1}\mathbf{X})E\{\mathbf{m}_n^{Nt}\}. \tag{5}$$

Since $\mathbf{X}_n^{-1}\mathbf{X} \approx I$, the eigenvalue spread of input correlation matrix $\mathbf{X}$ does not affect the convergence of geometric progression $E\{\mathbf{m}_n^{Nt}\}$ to zero. It shows that the convergence characteristics of the LMS-Newton algorithm are independent of the eigenvalue spread of $\mathbf{X}$, and therefore its rate of convergence is predictable. But it has an increased complexity of computing the inverse of input correlation matrix in each iteration, which makes it a comparatively less favorable choice.

## Normalized LMS Algorithm

In normalized LMS (NLMS) algorithm, we use a regularized inverse of $\mathbf{X}$ to precondition the input signals. Since $\mathbf{a}_n\mathbf{a}_n^T$ has rank one, it has at most one nonzero eigenvalue, given by: $\|\mathbf{a}_n\|_2^2 = \mathbf{a}_n^T\mathbf{a}_n$. This value can be very close to zero for numerically small values of input signals, in which case the algorithm might get unstable. In order to avoid division by zero, we choose $(\psi I + \mathbf{a}_n\mathbf{a}_n^T)^{-1}$ as a regularized inverse of $\mathbf{a}_n\mathbf{a}_n^T$, where $\psi \approx 0$. The update equation is then given by:

$$\mathbf{w}_{n+1}^{Nm} = \mathbf{w}_n^{Nm} + \mu e(n)(\psi I + \mathbf{a}_n\mathbf{a}_n^T)^{-1}\mathbf{a}_n . \quad (6)$$

Using matrix inversion lemma, it becomes: $\mathbf{w}_{n+1}^{Nm} = \mathbf{w}_n^{Nm} + \dfrac{\mu}{\psi + \|\mathbf{a}_n\|_2^2} e(n)\mathbf{a}_n$.

The misalignment $\mathbf{m}_n^{Nm} = \mathbf{w}_n^{Nm} - \mathbf{w}_{opt}$ of NLMS-tap weight vector must satisfy the relation:

$$E\{\mathbf{m}_{n+1}^{Nm}\} = \left( I - \frac{\mu}{\psi + \|\mathbf{a}_n\|_2^2}\mathbf{X} \right) E\{\mathbf{m}_n^{Nm}\} . \quad (7)$$

Choice of stepsize parameter $\mu^{Nm} = \mu \big/ \left(\psi + \|\mathbf{a}_n\|_2^2\right)$ offers a good tradeoff between fast convergence and good tracking abilities, but this algorithm has a drawback of increased misadjustment.

## TD-LMS Algorithm

Transform domain LMS algorithms is a class of robust preconditioned algorithms having good tracking capabilities in non stationary environments. Application of an orthogonal transform (e.g. FFT, DCT, or DST etc.) , followed by a power normalization step, has the ability to reduce the eigenvalue spread of input correlation matrix, which results in an increase of convergence speed of the algorithm[3]. The input vector $\mathbf{a}_n$, and weight vector $\mathbf{w}_n$ are transformed to $\hat{\mathbf{a}}_n = \mathrm{T}\mathbf{a}_n$ and $\hat{\mathbf{w}}_n = \mathrm{T}\mathbf{w}_n$ respectively, through an orthogonal transform $\mathrm{T}$. With error estimate $e(n) = s(n) - \hat{\mathbf{w}}_n^T\hat{\mathbf{a}}_n$, and power $\sigma_n(i)^2 = \beta\sigma_{n-1}(i)^2 + (1-\beta)\hat{\mathbf{a}}_n(i)^2$; $\forall\ i = 0,1,\ldots,N-1$, where $0 < \beta < 1$, the weight vector update equation is:

$$\hat{\mathbf{w}}_{n+1} = \hat{\mathbf{w}}_n + 2\mu D_n^{-1} e(n)\hat{\mathbf{a}}_n . \quad (8)$$

Here $D_n = diag\left(\sigma_n(0)^2, \sigma_n(1)^2, \ldots, \sigma_n(N-1)^2\right)$ is a diagonal matrix consisting of the approximated eigenvalues of $\mathbf{X}$. TD-LMS algorithm follows an almost straight path to the optimal solution and that the effect of the eigenvalue spread is compensated by the power normalization.

## Affine Projection Algorithm (APA)

All the algorithms, discussed in previous sections, update the weight vector on the basis of a single input vector and can be viewed as one dimensional affine projection algorithms. Affine projection algorithm (APA) is a generalization of the NLMS algorithm in multiple dimensional affine spaces. APA uses data-reusing technique to have an increased convergence speed of under correlated input signals[5]. This data reusing is done by keeping the last $L+1;(L\geq 0)$ input vectors in a matrix $\mathbf{A}_n^{AP}=[\mathbf{a}_n,\mathbf{a}_{n-1},\ldots,\mathbf{a}_{n-L}]$. With definitions

- desired signal vector $\mathbf{d}_n^{AP}=[s(n),s(n-1),\ldots,s(n-L)]$,
- filter output vector $\mathbf{y}_n^{AP}=\mathbf{A}_n^{AP\,T}\mathbf{w}_n^{AP}=[y(n),y(n-1),\ldots,y(n-L)]$, and
- apriori error vector $\mathbf{e}_n^{AP}=\mathbf{d}_n^{AP}-\mathbf{y}_n^{AP}=[e(n),e(n-1),\ldots,e(n-L)]$,

the update equation of affine projection algorithm, as realized from Eq-(6) is:

$$\mathbf{w}_{n+1}^{AP}=\mathbf{w}_n^{AP}+\mu\,\mathbf{A}_n^{AP}(\psi I+\mathbf{A}_n^{AP\,T}\mathbf{A}_n^{AP})^{-1}\mathbf{e}_n^{AP}\,. \tag{9}$$

The affine projection algorithm is a better alternative than NLMS in applications where the input signal is highly correlated. The convergence speed of the tap weight vector increases with an increase in projection dimension, and unfortunately, the algorithm's computational complexity increases too[6] .

## COMPARATIVE PERFORMANCE IN SYSTEM IDENTIFICATION

In order to compare the performance of adaptive filters, described in previous section, we use the application of unknown system identification. The block diagram of the model is shown in Figure 1 [1].

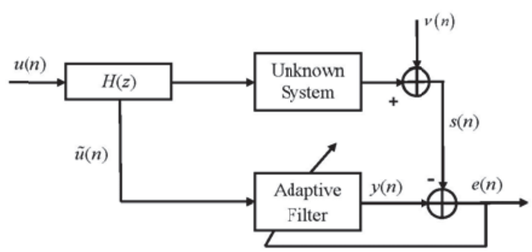


**FIGURE 1.** Block diagram of Adaptive system identification (ASI) model.

Filter input/output sequence $\{\mathbf{a}_n,s(n)\}$ is obtained by passing white Gaussian signal $u(n)$ of variance $\sigma^2=1$, through a coloring filter with frequency response : $H(z)=\frac{\sqrt{1-\alpha^2}}{1-\alpha z^{-1}}$, where $|\alpha|<1$, $\alpha$ is the correlation parameter and controls the spectral properties of input signal's autocorrelation matrix. $\alpha=0$ corresponds to the case when eigenvalue spread is close to 1, and eigenvalue spread increase with an increase in the value of $\alpha$. An independent Gaussian noise $v(n)$ of variance 0.001 is added in the output signal $s(n)$, at instant $n$. The performance of all the algorithms is measured by minimizing the normalized misalignment (in dB), $10\log_{10}\left(\|\mathbf{w}_n-\mathbf{w}_{opt}\|_2/\|\mathbf{w}_{opt}\|_2\right)$.

For all the simulations, initial tap-weight vector is taken as $\mathbf{w}_o(i)=1\ \forall\ i=1,\ldots,N$, while a different value of stepsize parameter is chosen for each algorithm, so as to maintain their stability, and these values are : $\mu_{lms}=0.015$, $\mu_{Nt}=0.06$, $\mu_{Nm}=0.3$, $\mu_{TD}=0.06$, $\mu_{APA}=0.3$. All the simulations are performed for 1000 iterations, with an ensemble average of 200 independent runs. Convergence of misalignment is shown in Table 1 for $\alpha=0.5\,\&\,0.85$, and $N=5\,\&10$. It is clear from Table 1, and Figure 2, that although LMS algorithm has slow convergence rate, but it is robust and exhibit good convergence of misalignment to zero.

**TABLE(1).** Norm-2 of the Misalignment of iterative algorithms for $n = 1000$ .

| **Algorithm** | $\alpha = 0.0, N = 5$ | $\alpha = 0.5, N = 5$ | $\alpha = 0.5, N = 10$ | $\alpha = 0.85, N = 5$ | $\alpha = 0.85, N = 10$ |
|---|---|---|---|---|---|
| LMS | 0.005738 | 0.009859 | 0.010155 | 0.026207 | 0.010609 |
| LMS-Newton | 0.013175 | 0.021317 | 0.020108 | 0.012811 | 0.067711 |
| NLMS | 0.010316 | 0.023599 | 0.015334 | 0.022428 | 0.035410 |
| TD-LMS | 0.014521 | 0.022428 | 0.024578 | 0.009403 | 0.047362 |
| APA | 0.020215 | 0.035195 | 0.021997 | 0.022866 | 0.0426390 |

As far as performance of remaining four algorithms is concerned, it can be seen in Figure 2(a-d) that they have failed to show better convergence of misalignment. Table 1 shows that for smaller filter length and small values of $\alpha$ , LMS-Newton and TD-LMS algorithms have almost the same misalignment near $n = 1000$ , while misalignment of LMS-Newton gets poor with an increase in value of $\alpha$ , as compared with its variant TD-LMS algorithm. An overall comparison of LMS-Newton, NLMS, TD-LMS and APA shows preferred misalignment behavior of NLMS algorithm. APA has fastest convergence rate of MSE among these algorithms, but has poorest convergence of misalignment. Figures 2(a) and (c) compare the normalized misalignment for $N = 5$ with correlated signals corresponding to $\alpha = 0.5$ & $\alpha = 0.85$ respectively. In both cases TD-LMS exhibit closest normalized convergence to LMS, while learning curves of normalized misalignment in Figure 2(b)&(d) for $N = 10$ show preference of NLMS algorithm in normalized misalignment behavior.

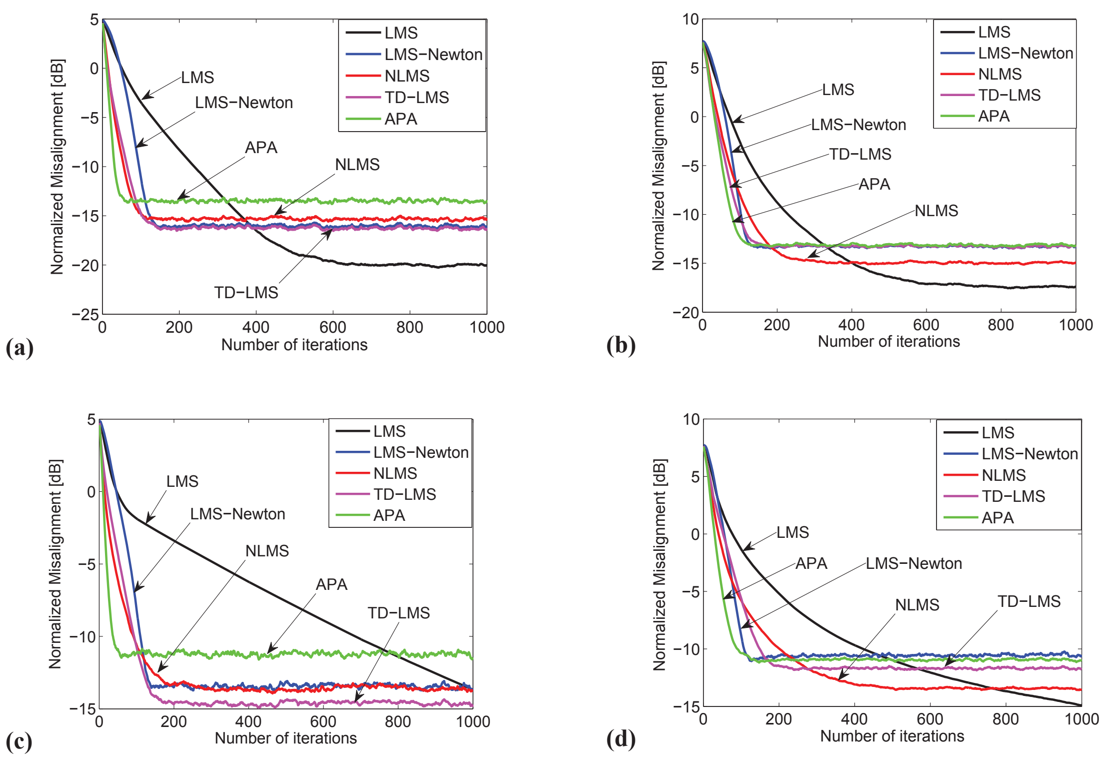


**FIGURE 2.** Learning Curves of Normalized Misalignment for (a). $\alpha = 0.5$, and $N = 5$, (b). $\alpha = 0.5$, and $N = 10$, (c). $\alpha = 0.85$, and $N = 5$, (d). $\alpha = 0.85$, and $N = 10$.

In Figure 3, $N = 20$ and learning curves of normalized misalignment are recorded for 2000 iterations, with an ensemble average of 200 independent runs, and $\alpha = 0.85$. More iterations are considered because of slow convergence in MSE of LMS algorithm, with an increase in filter length, slowdowns convergence of normalized misalignment as well and to have a clear picture of observations we need to consider more time samples. This simulation with larger filter length presents a better comparison of misalignment behavior of all the algorithms. This comparison again shows the preference of NLMS algorithms over other invariants of LMS algorithm.

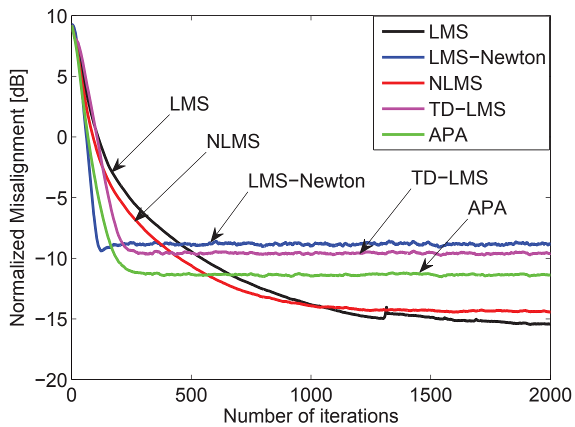


**FIGURE 3.** Learning Curves of Normalized Misalignment for $\alpha = 0.85$, and $N = 20$.

## CONCLUSION

The objective of this study was to compare the performance of well known modifications of LMS algorithm and find out the rooms of further improvement. The unknown system identification problem is solved by using LMS algorithm and its variants LMS-Newton, NLMS, TD-LMS and APA, and experiments are performed for observing learning curves of normalized misalignment and transient performances employing these algorithms. This comparison has shown that although APA has fast rate of convergence in MSE, however its misadjustment and convergence in misalignment are not good enough. On the other hand NLMS algorithm has normalized misalignment closest to that of LMS algorithm and remains stable, but its misadjustment needs improvement. This comparison highlights the need of designing new preconditioning techniques to develop variants of LMS algorithm, having better misadjustment and misalignment.

## ACKNOWLEDGMENTS

The authors highly acknowledge the support of the School of Mathematical Sciences, Universiti Sains Malaysia for accepting this paper.

## REFERENCES

1. N. A. Ahmad, *In Proceedings of International Conference on Applied Mathematics*, pp. 509-518, ( 2005).
2. B. Widrow and M. Kamenetsky, *Neural Networks* **16** (5), 735-744, (2003).
3. B. Farhang-Boroujeny, *Adaptive filters: theory and applications*, John Wiley & Sons, Inc., 1998.
4. S. Haykin, *Adaptive Filter Theory*, second ed., Englewood Cliffs, NJ: Prentice-Hall., 1991.
5. P. S. R. Diniz, *Adaptive filtering: algorithms and practical implementation*, Springer Verlag, 2008.
6. S. Narayan, A. Peterson and M. Narasimha, *Acoustics, Speech and Signal Processing, IEEE Transactions on* **31** (3), 609-615 (1983).
7. S. G. Sankaran and A. Beex, *In Proceedings of Thirty-First Asilomar Conf. Signals, Syst., Comput.,* pp.1670-1673, (1997).
8. J. Benesty, H. Rey, L. R. Vega and S. Tressens, *Signal Processing Letters, IEEE* **13** (10), pp. 581-584, (2006).
9. S. G. Sankaran and A. L. Beex, *Signal Processing, IEEE Transactions on* **48** (4), pp. 1086-1096, (2000).
10. E. Kreyszig, *Introductory functional analysis with applications*, Wiley, 1989.