

Discrete-time Variance Tracking with Application to Speech Processing

T.J.Moir

I.I.M.S., Massey University Albany Campus, Auckland, New Zealand

t.j.moir@massey.ac.nz

Abstract

Two new discrete-time algorithms are presented for tracking variance and reciprocal variance. The closed loop nature of the solutions to these problems makes this approach highly accurate and can be used recursively in real time. Since the Least-Mean Squared (LMS) method of parameter estimation requires an estimate of variance to compute the step size, this technique is well suited to applications such as speech processing and adaptive filtering.

1.Introduction

The algorithms in this paper can be applied to any problem where variance of a stochastic process is required to be accurately tracked with time. There are numerous such applications and this paper considers one such application related to speech and adaptive signal processing. At the heart of many adaptive algorithms is the least-mean squared (LMS) algorithm which is used for parameter (weight) estimation. The LMS algorithm can be applied to a number of specific problems but here (without losing generality) the identification of an unknown system driven by white noise is considered. For convenience a system represented as a finite impulse response (FIR) filter of order n is chosen. Consider the block diagram shown in Figure 1 where the unknown system is to be identified by adapting the weight vector W_k to minimise the variance of the error e_k .

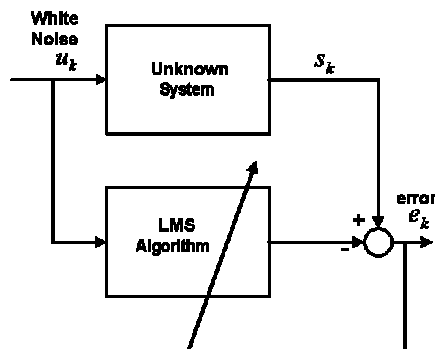


Figure 1. Identification using LMS

The input sequence $\{u_k\}$ is assumed to be zero-mean white Gaussian with variance σ_u^2 . The ordinary LMS algorithm can be written as [1]

$$e_k = s_k - U_k^T W_{k-1} \quad (1)$$

$$W_k = W_{k-1} + 2\mu U_k e_k$$

(2)

where the weight vector $W_k = [w_0 \ w_1 \ \dots \ w_n]^T$ and U_k is the vector of regressors (or tap vector) given by $U_k = [u_k \ u_{k-1} \ u_{k-2} \ \dots \ u_{k-n}]^T$, n is the system order with $n + 1$ weights. The sequence s_k is the output of the system to be identified and can be written as $s_k = U_k^T W_k^*$. Here W_k^* represents the vector of $(n + 1)$ system parameters (or optimal weight vector at time k). In theory, the condition for convergence in the mean square of the LMS algorithm [2] is that the step size

$$\mu < \frac{1}{(n+1)\sigma_u^2} \quad (3)$$

From which results in $W_k \rightarrow W_k^*$. The well known condition which involves the reciprocal of the largest eigen-value of the correlation matrix (convergence in the mean of the weight vector) is also automatically satisfied by (3).

2. Non Stationary signals

When $\{u_k\}$ is no longer white or stationary, equation (3) must be computed for the largest variance of σ_u^2 and some factor of safety must be provided to avoid instability in the LMS algorithm. For example, if the step size μ is overestimated then the LMS algorithm can become unstable but if it is underestimated then the convergence of the algorithm will be slow. Because of this, usually for non-stationary problems an attenuating factor for equation (3) has often been proposed. For example Feuer & Weinstein [3] show using statistical arguments that μ should be chosen to be one third of the commonly quoted maximum whilst Horowitz & Senne, [4] give the value as one sixth. To overcome this problem the variance is often computed and the step size varied with time. The variance can be computed at the Nth iteration either in a batch computation as in

$$\sigma_N^2 = \frac{1}{N} \sum_{i=1}^N u_i^2 \quad (4)$$

or as the recursive equivalent of (4) as given by Young [5]

$$\sigma_N^2 = \sigma_{N-1}^2 + \frac{1}{N} [u_N^2 - \sigma_{N-1}^2] \quad (5)$$

For stationary signals the recursive variance estimator in (5) above converges asymptotically as $N \rightarrow \infty$. However, if the signal is non-stationary then (5) is required to track the variance with time. When N becomes large, equation (5) pays little attention to new data and does not track well. It is usual in the literature to use some form of exponential weighting of past data to track the variance. This facilitates the use of an exponential ‘forgetting factor’. It is often proposed to use simple autoregressive estimators [6]

$$\sigma_N^2 = \lambda \sigma_{N-1}^2 + (1 - \lambda) u_N^2 \quad (6)$$

where $0 < \lambda < 1$ is the forgetting factor which controls the bandwidth and hence the time constant of the first order recursive digital filter of (6) above. For a typical speech signal, using $\lambda = 0.95$ can be made to work although the estimate is not very smooth. Increasing λ gives smoother estimates at the expense of worse tracking. The implicit approach in [7] uses a forgetting factor approach similar to the above. Whilst methods like these can be made to work for certain applications, they are generally of an *ad-hoc* nature and are a compromise between smoothness of the estimate and tracking ability. The approach used here can be used for accurate tracking of variance or to accurately define the variance of a signal with a pre-defined set-point. The philosophy is similar to that used in radio receivers where a low bandwidth non-linear AGC boosts the radio frequency signal to a useful power for later amplification and detection. Hence this approach is proposed as a front end to adaptive algorithms rather than an implicit change to the LMS algorithm itself, such as has been proposed by [7] An automatic gain control strategy has been proposed in [8] which improves the performance of an LMS adaptive filter but it too uses forgetting factors and is highly non-linear.

3. Automatic Variance Control and Estimation Loops

The automatic variance control (AVC) described here is essentially a form of AGC calibrated to variance rather than amplitude. Whilst it would seem obvious that an AGC may well suffice, it should be recalled that AGCs are largely non-linear and would not result in accurate tracking. With a few exceptions, AGCs do not include a pure integrator, instead relying on a simple low pass filter and several stages of amplifier gain. AGCs use voltage-controlled amplifiers instead of the linear multiplier used here and use some form of absolute value (diode rectification) instead of a pure squarer. The AVC is entirely linear with theoretically zero steady-state error to a step change in variance [9] and works down to white noise levels.

3.1 Automatic Variance Controller

The block diagram of the continuous-time AVC is shown in Figure 2. It consists of a pure squarer, a linear multiplier, a summing junction (with set-point) and an integrator.

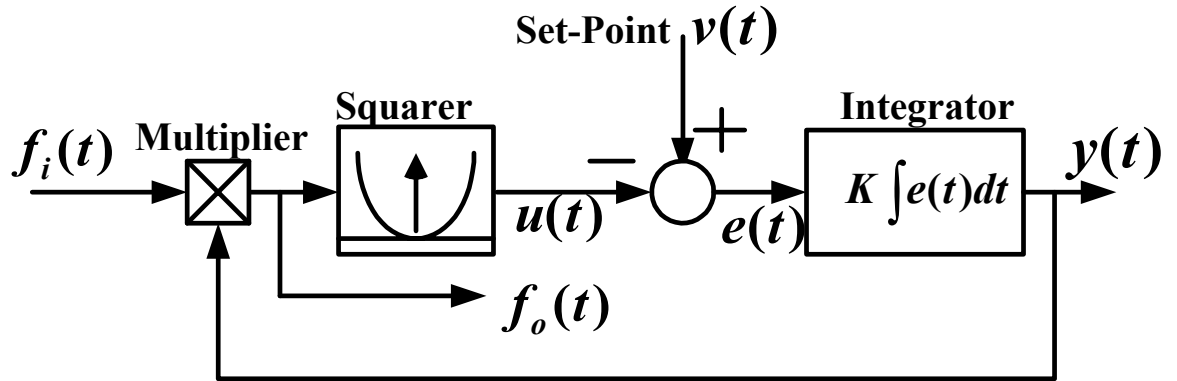


Figure 2. Automatic Variance Control

The operation of the circuit is essentially as follows. The measured input signal $f_i(t)$ is assumed to be dc free, but this is not a general restriction. This signal is multiplied by the integrator output $y(t)$ which for a stationary or periodic input signal will be constant (in steady-state) and for non-stationary inputs will be time-varying. The multiplier output is the output signal with the pre-defined variance defined by a set-point $v(t) > 0$. Should the input signal power change, the input to the squarer will also change momentarily and this is in turn fed into the squarer. When a signal is squared it produces a dc term plus higher harmonics. The higher harmonics are filtered by the integrator and any change in dc from the squarer output produces either a larger or smaller error from the summing junction which is in turn integrated. The integrator will either ramp up or down to scale the input signal to a pre-defined amount defined by $v(t)$. To provide suitable operation for variance rather than amplitude, a pure squarer rather than a modulus detector (normally used in AGCs) is used. It has been shown in [9] that the differential equation of the loop is given by

$$\frac{1}{K} \frac{dy(t)}{dt} = v(t) - (f_i(t)y(t))^2 \quad (7)$$

For high gain $K \gg 1$ and it can be shown from (7) that if the integrator output $y(t)$ is squared by a squarer outside the loop, then

$$y^2(t) \rightarrow \frac{v(t)}{E[f_i^2(t)]} \quad (8)$$

Hence for any non-zero signal present at $f_i(t)$, and for unity set-point, the integrator output squared will converge to the *reciprocal* of the variance of $f_i(t)$. In fact (8) above also holds for $f_i(t)$ signals with a dc component except it is the reciprocal of the average power of $f_i(t)$ rather than the reciprocal of variance. It can also be shown [9] that the multiplier output satisfies

$$E[f_o^2(t)] = v(t) \quad (9)$$

That is, the variance (or average power for signal with dc component) of the first multiplier output is controlled by the set-point.

If the bandwidth of the loop is chosen to be high enough it is possible to track the reciprocal of power (or variance) accurately with time. Clearly this information can be used in the LMS algorithm as part of the computation of step size (equation (3)). It is interesting that in (8) the reciprocal power is estimated rather than the power itself. This is convenient for LMS applications in that the step size should be chosen inversely proportional to power as stated by equation (3). Similarly, the estimation of Kurtosis for a zero-mean signal requires the reciprocal squared of variance [6].

3.2 Variance Estimation Loop

In order for variance to be estimated rather than its reciprocal, the output of the first loop needs to be squared and fed into a second feedback control system. This second loop is a multiplier with high gain around it which acts (for high gain within the bandwidth of the loop) as a divider algorithm. Similar approaches have been used for decades in analogue electronics using operational amplifiers. The second loop is shown in Figure 3 below for the continuous-time case.

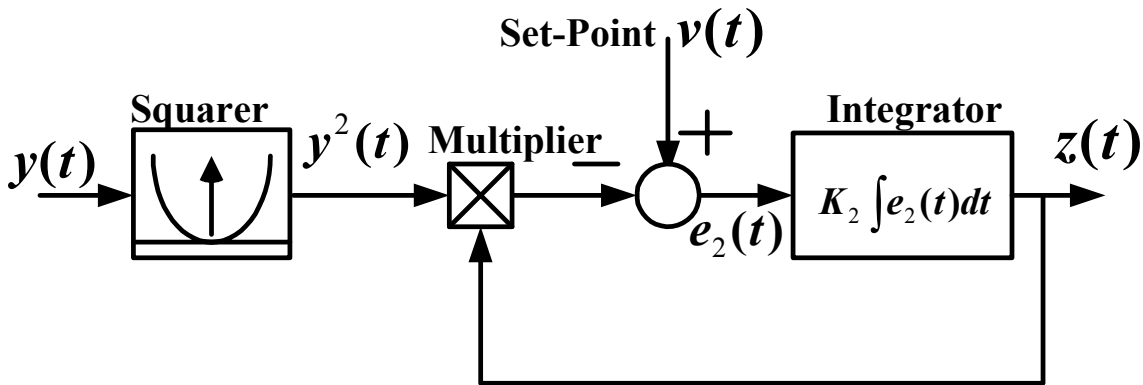


Figure 3 Analogue Variance tracking loop.

In Figure 3, $y(t)$ is fed from the integrator output shown in Figure 2. For such a system if high gain is used $K \gg 1$ and the error $e_2(t) \rightarrow 0$. If the error approaches zero then

$$v(t) - y^2(t)z(t) = 0 \quad (10)$$

giving

$$z(t) = \frac{v(t)}{y^2(t)} \quad (11)$$

which, when applying equation (8) becomes

$$z(t) = E[f_i^2(t)] \quad (12)$$

which is the estimate of variance.

4. Digital Variance Loops

Figure 2 is shown in analogue form but for this application it needs to be converted to discrete-time. First consider a discrete-time version of Figure 2. where v_k, y_k are respectively the discrete representations of the set-point and integrator output signals at sample instant k . This results in the following algorithm

4.1 Algorithm: Discrete Automatic Variance Control

Error Signal:

$$\mathbf{e}_k = \mathbf{v}_k - \mathbf{u}_k \quad (13)$$

Integrator Output:

$$\mathbf{y}_k = \mathbf{y}_{k-1} + \mathbf{K}\mathbf{e}_{k-1} \quad (14)$$

Multiplier Output:

$$\mathbf{f}_k^o = \mathbf{f}_k^i \mathbf{y}_k \quad (15)$$

Squarer Output:

$$\mathbf{u}_k = (\mathbf{f}_k^o)^2 \quad (16)$$

The discrete signals \mathbf{f}_k^o and \mathbf{f}_k^i above represent the sampled versions at sample instant k of the continuous time signals $\mathbf{f}_o(\mathbf{t})$ and $\mathbf{f}_i(\mathbf{t})$ respectively.

This algorithm behaves in the same way as the analogue version in steady-state. That is, after squaring (outside of the loop) the integrator output (14) we have

$$\mathbf{y}_k^2 \rightarrow \frac{\mathbf{v}_k}{E[(\mathbf{f}_k^i)^2]} \quad (17)$$

which indicates that provided the set-point is unity, the integrator output squared tracks the reciprocal variance of the measured (sampled) input signal \mathbf{f}_k^i . Also, it can be shown that the signal from the multiplier output (equation 15) will have an average variance pre-defined by the set-point \mathbf{v}_k . For most applications the variance itself is required and the next loop is also necessary [9].

4.2 Algorithm: Discrete Variance Estimation Loop

The discrete version of the Variance Tracking Loop of Figure 2 becomes

Error Signal:

$$\mathbf{l}_k = \mathbf{v}_k - \mathbf{m}_k \quad (18)$$

Integrator Output:

$$\mathbf{z}_k = \mathbf{z}_{k-1} + \mathbf{K}_2 \mathbf{l}_{k-1} \quad (19)$$

Multiplier Output:

$$\mathbf{m}_k = \mathbf{z}_k \mathbf{y}_k^2 \quad (20)$$

Where $\mathbf{l}_k, \mathbf{z}_k$ are respectively the error signal and the integrator output in the discrete-time case. The signal \mathbf{y}_k^2 is taken from the squared (external) integrator output of the first loop. Provided both loops remain stable, the above algorithms will result in

$$\mathbf{z}_k = E[(\mathbf{f}_k^i)^2] \quad (21)$$

4.3 Biasing the Input Signal

Aside from stability, which is discussed in 5, algorithm 4.1 may encounter numerical problems if the input signal \mathbf{f}_k^i becomes small. This is because the integrator output squared is proportional to reciprocal variance (equation 17) and hence will become large and divergent as the input signal becomes small. In many cases the low-level background white noise is enough to prevent this problem but nevertheless it

has been encountered several times with a real-time implementation. If variance is to be estimated instead of reciprocal variance then an offset can be added to the input signal to provide a signal input which is always suitably scaled to be non-zero. The input signal now becomes $f_c + f_k^i$. The

corresponding squared integrator output of algorithm 4.1 will be $y_k^2 \rightarrow \frac{v_k}{f_c^2 + E[(f_k^i)^2]}$ and when

this is in turn fed into algorithm 4.2 we arrive at an overall variance estimate of $z_k = f_c^2 + E[(f_k^i)^2]$. This represents dc power plus ac power. If the original bias squared (or dc power) is subtracted from this then the true variance is found. This biasing technique worked well in practice. However, if reciprocal variance is required (as with the LMS algorithm) then a further subsequent division is required. Note here that it is still assumed that the input signal is still assumed zero-mean even though it is biased.

5 Stability of AVC Loops

In the case of the analogue AVC loop it consists of a continuous time integrator with feedback consisting of a time-varying gain. Theoretically the analogue loop can never be unstable since an integrator with negative feedback is always stable. Of course in a real implementation there are other considerations such as the limitations of operational amplifiers which define the integrator and so on. Parasitic oscillations can occur if care is not taken with the analogue design. However, in its raw form the AVC is always stable in continuous time. For discrete time the stability is limited and can be derived as follows.

5.1 Stability of Discrete Automatic Variance Control Loop

To find the stability of algorithm 4.1. Substitute the error of equation (13) into the expression for the integrator output (14) giving

$$y_k = y_{k-1} + K[v_{k-1} - u_{k-1}] \quad (22)$$

After further substitutions of (15) and (16) into (22) we arrive at

$$y_k = y_{k-1} - K(f_{k-1}^i)^2 y_{k-1}^2 + K v_{k-1} \quad (23)$$

And we are required to find the upper bound on the gain K for constant set-point $v_k = 1$ We consider three cases

Case I

Constant Input signal $f_k^i = f_c$. The condition for static equilibrium is at steady state

$$y_k = y = \frac{1}{f_c} = -\frac{1}{f_c} \quad (24)$$

Write (23) above in the form

$$y_k = g(y_{k-1}) \quad (25)$$

and following the procedure in [10] determine

$$g'(y_{k-1}) = 1 - 2Kf_c y_{k-1} \quad (26)$$

Substituting (24) gives

$$g'(y) = 1 - 2K \quad (27)$$

and

$$g'(y) = 1 + 2K \quad (28)$$

For asymptotic stability we require

$$|g'(y)| < 1 \quad (29)$$

which results in

$$-1 < Kf_c < 1 \quad (30)$$

Case II

Constant input plus small complex frequency perturbation $f_k^i = f_c + \epsilon e^{j\theta n}$

Look for solutions of the form

$$y_k = \frac{1}{f_c} + \epsilon z_n e^{j\theta n} + \eta(\epsilon^2) \quad (31)$$

and

$$y_k = -\frac{1}{f_c} - \epsilon z_n e^{j\theta n} - \eta(\epsilon^2) \quad (32)$$

Substituting (31) and (32), (ignoring higher order terms) into (23) results in two new difference equation

$$z_k = e^{-j\theta} z_{k-1} - e^{-j\theta} 2K(z_{k-1} f_c + \frac{1}{f_c}) \quad (33)$$

and

$$z_k = e^{-j\theta} z_{k-1} + e^{-j\theta} 2K(z_{k-1} f_c + \frac{1}{f_c}) \quad (34)$$

The above two equations can be written in the form

$$z_k = \mathbf{g}(z_{k-1}) \quad (35)$$

For asymptotic stability [10]

$$|\mathbf{g}'(z_{k-1})| < 1 \quad (36)$$

which gives the range

$$-1 < K f_c < 1 \quad (37)$$

which is the same result as a constant input in Case I above. For a sine wave $\sin(\theta)$ its squared expansion is $\frac{1}{2}[1 + \cos(2\theta)]$. The $\sin(2\theta)$ term can be neglected since it is a ripple term which is filtered attenuated by the integrator. Hence $f_c^2 \approx \frac{1}{2}$ and $f_c \approx \frac{1}{\sqrt{2}}$ with (from 37) the resulting bounds on stability as $-\sqrt{2} < K < \sqrt{2}$.

5.2 Stability of Discrete Variance Estimation Loop

In a similar manner to the above the integrator output in algorithm 4.2 is

$$z_k = z_{k-1} + K_2[v_{k-1} - y_{k-1}^2 z_{k-1}] \quad (38)$$

The above time-variant difference equation can be written as

$$z_k = \beta_{k-1} z_{k-1} + K_2 v_{k-1} \quad (39)$$

where

$$\beta_{k-1} = 1 - K_2 y_{k-1}^2 \quad (40)$$

is a real time-varying scalar. Clearly for this scalar case the modulus of β_{k-1} must be less than unity for all k giving

$$0 < \mathbf{K}_2 < \frac{2}{\mathbf{y}_{k-1}^2} \quad (41)$$

Taking expectations of (39) and using (17) we obtain the statistical result

$$\mathbf{K}_2 < \frac{2E[(f_{k-1}^i)^2]}{\mathbf{v}_{k-1}} \quad (42)$$

Hence for unity set-point the upper bound on \mathbf{K}_2 is

$$\mathbf{K}_2 < 2\sigma_i^2 \quad (43)$$

6. Illustrative Example

To illustrate the new method consider a 4 second speech signal passed through an FIR system with 10 constant and assumed unknown weights as in Figure 1. Figure 3 shows the speech signal and the output of the variance estimator. For convenience, the squared output $y(t)$ of the variance controller has been reciprocated using algorithm 4.2 to show actual variance. The LMS algorithm does not of course require this step but for a robust estimate avoiding numerical problems it is best to estimate variance using the biasing technique of section 4.3 above. Figure 4 shows respectively the mean square errors of the LMS algorithm for the new method and for conventional LMS. For the new method, the step size (3) of the LMS algorithm is computed at each iteration based on a measure of variance reciprocal from the new algorithm whilst for conventional LMS a step size is used based on the maximum possible variance.

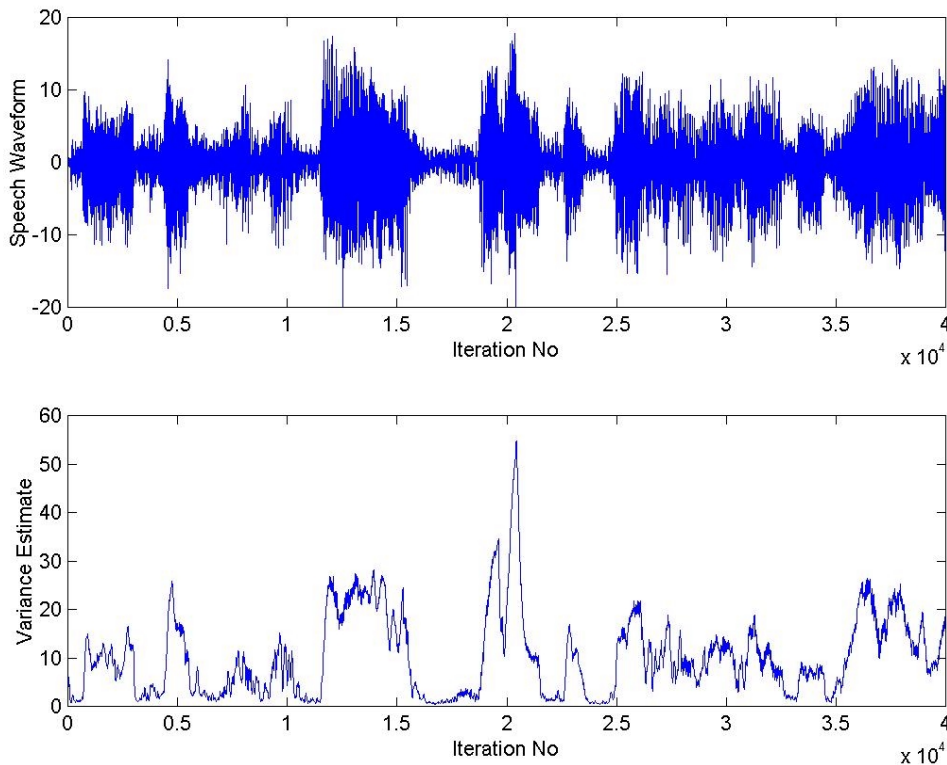


Figure 3. Speech Signal and Variance Estimate

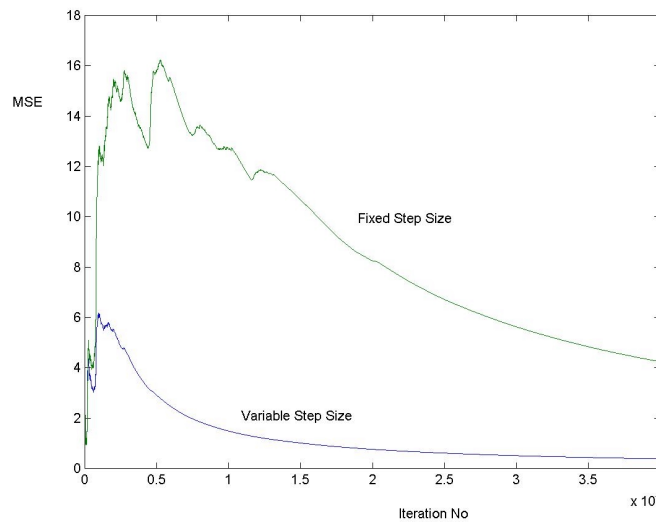


Figure 4. Comparison of Mean Squared Errors (MSE) for fixed and variable step sizes.

It can be seen from Figure 4 that by updating the LMS algorithm at each iteration the rate of convergence of the LMS algorithm is dramatically improved and this gives rise to a smaller mean-square weight error. Note also the smoothness of the variance estimate in Figure 3.

6. Conclusion

Two new discrete-time algorithms for accurately tracking both the reciprocal of variance versus time and variance versus time have been described. The first of these algorithms also has also the property that the variance of a time-varying input signal can be finely controlled by a set-point. When used in tandem with the LMS algorithm the mean-square error of the LMS algorithm can be significantly reduced. Whilst it is not claimed that the method of updating the LMS step size at each iteration is in any way novel, the application of these new variance estimation techniques has not been pursued elsewhere. The variance estimators will track step changes in variance with zero steady-state error. For signals whose variance varies in the form of a ramp or higher-order polynomial form, these algorithms will give a finite, but small steady-state error. To overcome this difficulty a second integrator with appropriate phase-lead stabilisation could be added to algorithm 4.1 at the expense of greater stability problems. The algorithms can be applied to any application where the signal is directly measurable and are simple to implement. A further development of the algorithms would be to change them so as to estimate higher moments such as Kurtosis [6].

7. References

- [1] Widrow, B. & Stearns, S D. (1985) "Adaptive Signal Processing", Prentice Hall, Englewood Cliffs, NJ.
- [2] Haykin, S. (1986) "Adaptive filter theory", Prentice Hall Englewood Cliffs, New Jersey
- [3] Feuer, A & Weinstein, E. (1985) "Convergence Analysis of LMS Filters with Uncorrelated Gaussian Data", IEEE Trans ASSP, Vol 33, No 1, pp 222-229
- [4] Horowitz, L L & Senne, K. (1981) "Performance Advantage of Complex LMS for Controlling Narrow-band Adaptive Arrays", IEEE Trans, ASSP, Vol 29, No 3, pp 722-736
- [5] Young, P.C. (1984), "Recursive Estimation and Time Series Analysis: An Introduction". Springer Verlag: Berlin.
- [6] De Leon, P, "Short time Kurtosis of Speech Signals with application to co-channel speech separation", Proceedings of International Conference on Multimedia and Expo. New York, NY, USA. 30 July-2 Aug. 2000.
- [7] Shan, T.J. & Kailath, T. (1988), Adaptive algorithms with an automatic gain control feature, IEEE Transactions on Circuits & Systems, vol.35 no.1, pp122-127

- [8] Tapia J. Kuo S.M. (1990), "An optimized automatic gain controller for real time recursive LMS adaptive filters", IEEE International Symposium on Circuits and Systems (Cat.No.90CH2868-8), pp.3154-7 vol.4. New York, NY, USA.
- [9] Moir,T.J, (2001), "Automatic Variance Control", Circuits,Systems and Signal Processing, Birkhauser Boston ,vol 20,no 1,pp1-10
- [10] Alaydi,S.N.,(1996),"An Introduction to Difference Equations",Springer-Verlag

Acknowledgements

The author would like to thank Prof James Sneyd of Massey University for his analysis of stability in section 5.1 and for helpful comments by Dr J.F.Barrett.