CIRCUITS SYSTEMS SIGNAL PROCESS
VOL. 20, NO. 1, 2001, PP. 1–10

AUTOMATIC VARIANCE CONTROL AND VARIANCE ESTIMATION LOOPS*

T. J. Moir[1]

Abstract. A closed-loop servo approach is applied to the problem of controlling and estimating variance in nonstationary signals. The new circuit closely resembles, but is not the same as, automatic gain control, which is common in radio and other circuits. The closed-loop nature of the solution to this problem makes this approach highly accurate, and it can be used recursively in real time.
Key words: Automatic gain control, variance estimation, closed-loop control, adaptive filters.

1. Introduction

In applications that use adaptive filters, usually some estimate of variance is required if a least-mean squares (LMS) algorithm is used for weight vector estimation [1]. Normally, a window or moving window of data can be used, and the sample variance computed. A finer approach is to update the variance recursively. That is (assuming large k), for a sample variance σ_k^2 of k samples of zero-mean data y_k

$$\sigma_k^2 = \frac{1}{k}\sum_{i=1}^{k} y_i^2, \tag{1}$$

the recursive equivalent is given by [4]

$$\sigma_k^2 = \sigma_{k-1}^2 + \frac{1}{k}[y_k^2 - \sigma_{k-1}^2]. \tag{2}$$

For stationary signals, the recursive variance estimator converges asymtotically as $k \to \infty$. A similar method can be used for estimating the mean recursively [4].

* Received June 19, 2000.

[1] Institute of Information and Mathematical Sciences, Massey University at Albany, Private Bag 102-904, North Shore Mail Centre, Auckland, New Zealand. E-mail: t.j.moir@massey.ac.nz

However, if the signal is nonstationary, then (2) must be able to track the time-varying variance.

The problem is that with large k, equation (2) pays little attention to new information and "switches off." Sometimes a factor of safety is used, and the variance magnitude is underestimated to allow for any sudden changes, or if the dynamic range of the signal (as is the case with speech) is known, a worst-case upper estimate can be used. In the case of the LMS algorithm, if the variance is underestimated, the algorithm can become unstable due to the step size becoming too large. Conversely, if the variance is overestimated, the convergence of the LMS algorithm may well be too slow [1]. This basic problem has been recognized in [2], where the authors alter the LMS algorithm to implicitly include automatic gain control (AGC). 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 a "forgetting factor." For example, it is often proposed to use

$$\sigma_k^2 = \beta\sigma_{k-1}^2 + (1-\beta)y_k^2, \tag{3}$$

where $0 < \beta < 1$ is the forgetting factor, which controls the bandwidth and the time constant of the first-order recursive digital filter. For a typical speech signal, using $\beta = 0.95$ can be made to work, although the estimate is not very smooth. Increasing β gives smoother estimates at the expense of worse tracking.

The implicit approach in [2] uses a forgetting factor approach similar to the one above. Although methods like these can be made to work for certain applications, they are generally used ad hoc 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 better still to accurately define the variance of a signal with a predefined setpoint. The philosophy is similar to that used in radio receivers, where an 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 in [2]. An AGC strategy has been proposed in [3] which improves the performance of an LMS adaptive filter, but it too uses forgetting factors and is highly nonlinear.

2. Automatic variance control

The automatic variance control (AVC) described here is essentially a form of AGC calibrated to variance rather than amplitude or average power rather than voltage. Although it seems obvious that an AGC may well suffice, recall that AGCs are largely nonlinear and would not result in accurate tracking. Although the AVC uses nonlinear elements, its tracking ability is entirely linear with theoretically zero steady-state error to a step change in variance.

The block diagram of the AVC is shown in Figure 1. It comprises a pure squarer, a linear multiplier, a summing junction (with setpoint), and an integrator.

The operation of the circuit is essentially as follows. The input signal $f_i(t)$ is assumed to be free of dc. This signal is multiplied by the integrator output $y(t)$, which for a stationary or periodic input signal will be constant. The multiplier output is the output signal with the predefined variance defined by the setpoint $v(t)$. 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 predefined amount defined by $v(t)$. A squarer is used within the loop as opposed to the modulus detector normally used in AGC loops because the square provides the suitable scaling for variance rather than amplitude.

3. Steady-state analysis

The analysis follows from the block diagram of Figure 1. The input $f_i(t)$ can be either deterministic or stochastic but must be nonzero in the long term and free from dc. The integrator output becomes

$$y(t) = K \int e(t)\, dt. \tag{4}$$

The error signal $e(t)$ is defined as the difference between the setpoint $v(t)$ and the squarer output $u(t)$

$$e(t) = v(t) - u(t), \tag{5}$$

where $u(t)$ when combined with the multiplier output results in

$$u(t) = (f_i(t)y(t))^2. \tag{6}$$

The setpoint $v(t) > 0$ is normally a constant and can be set to unity for simplicity.

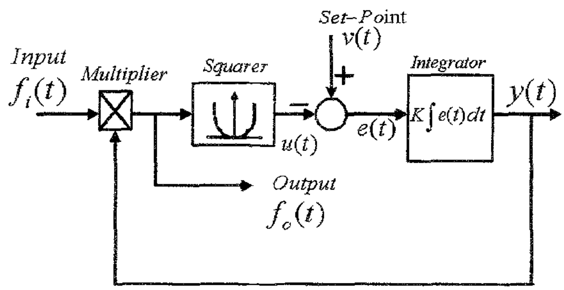

Figure 1. AVC block diagram.

Combining (4), (5), and (6) results in the differential equation

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

For any closed-loop system, the gain K must be as high as possible for a given bandwidth.

Assuming $K \gg 1$ in (7) results in

$$(f_i(t)y(t))^2 \to v(t). \tag{8}$$

However, because $f_o(t) = f_i(t)y(t)$, we must have

$$f_o^2(t) \to v(t). \tag{9}$$

This can be interpreted as follows for two classes of input signal.

3.1. Deterministic input signals

For a setpoint variance $v(t) = 1$ and input waveform $f_i(t)$ consisting of a sine wave of unity amplitude, the output of the AVC $f_o(t)$ will have a root mean square (rms) value of unity or an amplitude of $\sqrt{2}$. Provided there is no dc offset on the input waveform, the output waveform is always scaled so that the rms value of $f_o(t)$ follows the square root of the setpoint $v(t)$ from (9). For a sine wave input of frequency f_u Hz, the squarer output, from basic trigonometry, produces a dc term and a component at $2f_u$ Hz. Just as with a phase-locked loop, the $2f_u$ term must be sufficiently attenuated by the bandwidth of the loop to avoid distortion. This is achieved by ensuring that the unity gain crossover frequency of the integrator is at least 10 times smaller than the $2f_u$ component. Hence the bandwidth of the AVC must be chosen to be at least one-tenth of the lowest frequency of interest. A phase-locked loop can operate at much higher bandwidths because the input frequency is usually in the kHz or MHz region. Clearly for our applications, because the input frequencies are usually baseband, the AVC, like an AGC, is a slow acting control loop with a practical bandwidth of at most 25 Hz.

3.2. Random input signals

Taking expectations of both sides of (9) gives

$$E[f_o^2(t)] \to E[v(t)]. \tag{10}$$

The variance setpoint $v(t)$ is deterministic, and clearly $E[v(t)] = v(t)$.

Hence from (10) it is seen that the random output signal will be scaled so that its variance tracks the setpoint. For example, a zero-mean white noise input of unit variance and a setpoint variance of unity will result in an output variance of unity, and the statistical characteristics will remain otherwise unchanged. For the same setpoint, if the input variance slowly increases or decreases, the output variance will stay at unity.

For a time-varying input of, say, a speech waveform, the AVC will not have sufficient bandwidth to respond to periods of nonspeech. This is a positive result because any amplification of background noise is undesirable, as is any distortion of the speech "envelope."

4. Estimation of input variance

With slight modification, the AVC can be used as an accurate recursive method of estimating variance and will not suffer from the same disadvantages as discussed for equations (1) and (2). This method, like all closed-loop methods, is only limited by the bandwidth of the loop. A fundamental assumption is made here that the integrator output is statistically independent of the input signal, and we can write

$$E[f_o^2(t)] = E[f_i^2(t)]E[y^2(t)]. \tag{11}$$

This assumption is borne out by the fact that the bandwidth of the AVC is small, and the integrator smoothes out any high-frequency components. For stationary input signals, the integrator output will be a constant in steady state, and effectively $y(t) \to y_o$. A similar argument is used in the analysis of the weights for an LMS algorithm [1].

The input variance can now be computed by rearranging (11) and substituting the setpoint variance instead of $E[f_o^2(t)]$ via (10). Then

$$E[f_i^2(t)] \to \frac{v(t)}{E[y^2(t)]}, \tag{12}$$

which requires the division of the setpoint with the mean-squared integrator output. To avoid division, which is computationally time consuming, a further closed-loop system is used (see Figure 2), which consists of an integrator, a setpoint, and a multiplier.

An additional squarer is required outside the AVC loop from the integrator which obtains $y^2(t)$. The analysis of the loop of Figure 2 follows closely the method used previously. From the block diagram, the error signal is given by

$$e_2(t) = v(t) - y^2(t)y_2(t), \tag{13}$$

where $y_2(t)$is the integrator output in Figure 2.

Assume that the integrator gain $K \gg 1$; this results in

$$y_2(t) \to \frac{v(t)}{y^2(t)}, \tag{14}$$

which is the instantaneous division of the setpoint variance with the squared output of the AVC integrator. Taking expectations of (14) results in equation (12),

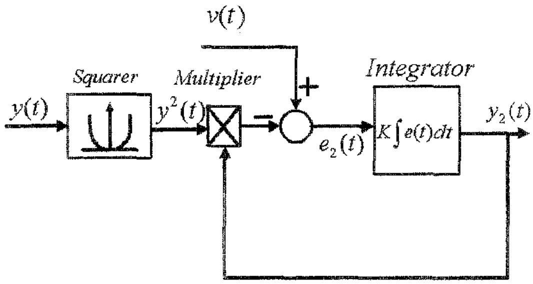

Figure 2. Variance estimation loop.

indicating that the integrator output of Figure 2 has a variance equal to the variance of the original input waveform. In steady state its value is constant and gives a direct reading of variance. For deterministic inputs, the variance estimator gives the rms value squared as an output or average power in the waveform. The same restrictions on bandwidth as the AVC do not apply because there is no feed-through component, hence the bandwidth of the integrator of this loop is chosen to be as much as 10 times faster than the AVC loop. It is also interesting to see by rearranging (12) that the output of the integrator of the AVC $y(t)$ has the form

$$E[y(t)] \to \frac{\sqrt{v(t)}}{\sigma}, \tag{15}$$

where σ is the standard deviation (or rms value) of the input signal (for a dc-free signal). For unity setpoint, (15) simplifies to the reciprocal of the standard deviation.

5. Simulation results

The AVC and variance estimation loops can be implemented either in digital or analog form. For this work, the loops were implemented in software, and several different signal types were tested. For convenience, a setpoint of unity was used for all cases. For a sine wave input of unity amplitude and unity setpoint, the AVC gave an output of a sine wave with magnitude 1.412, as expected. Figure 3a shows the original sinusoid, and Figure 3b shows the AVC output. The integrator output of the AVC is shown in Figure 3c which converges to 1.412. The sinusoid was chosen to have a frequency of 100 Hz, and the bandwidth of the AVC loop was 10 Hz. The ripple is due to 200 Hz feed-through terms caused by the squaring action within the loop.

The AVC was then tested on a speech signal. Two different bandwidths were used to illustrate the effect of too high a bandwidth. At first, speech sampled at 12 kHz was processed with an AVC bandwidth of 0.1 Hz. The setpoint variance was unity. Figure 4a shows the original speech signal, and Figure 4b shows the

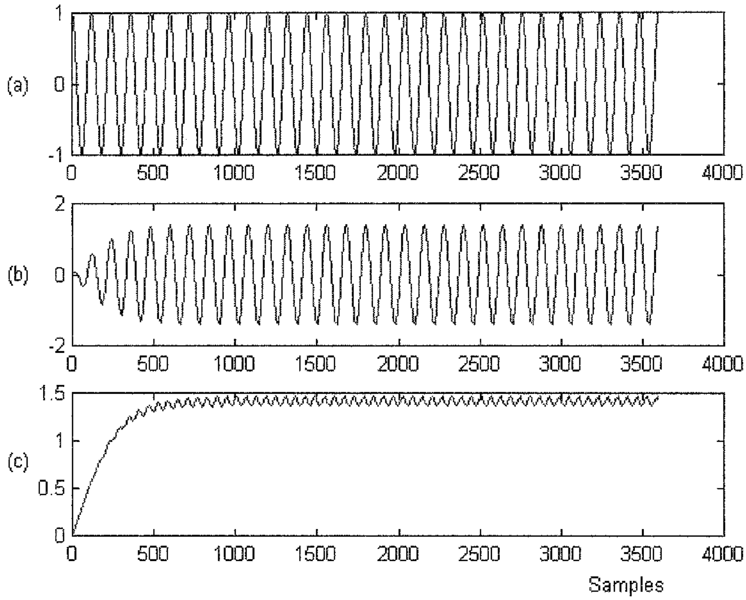

Figure 3. AVC acting on a sinusoid. (a) Input waveform to AVC $f_i(t)$, (b) AVC output $f_o(t)$, (c) integrator output $y(t)$.

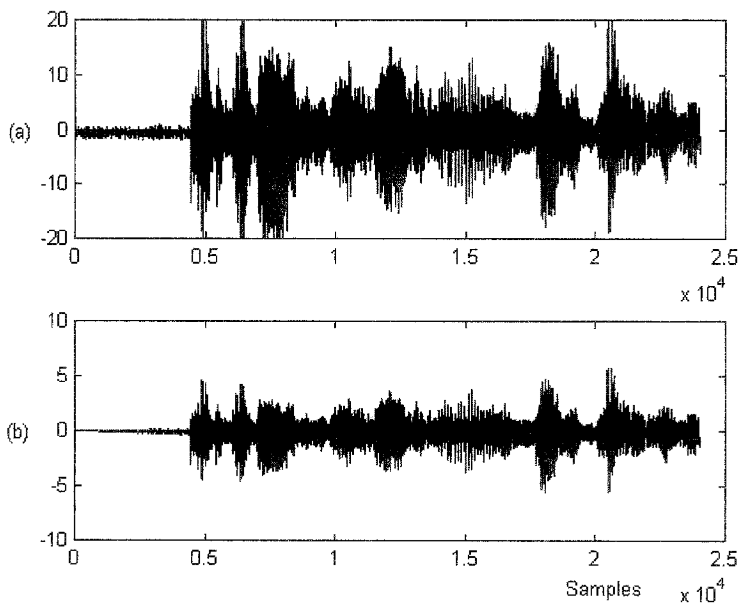

Figure 4. AVC output (bandwidth 0.1 Hz) for a speech waveform. (a) Speech waveform input to AVC$f_i(t)$, (b) output of AVC $f_o(t)$.

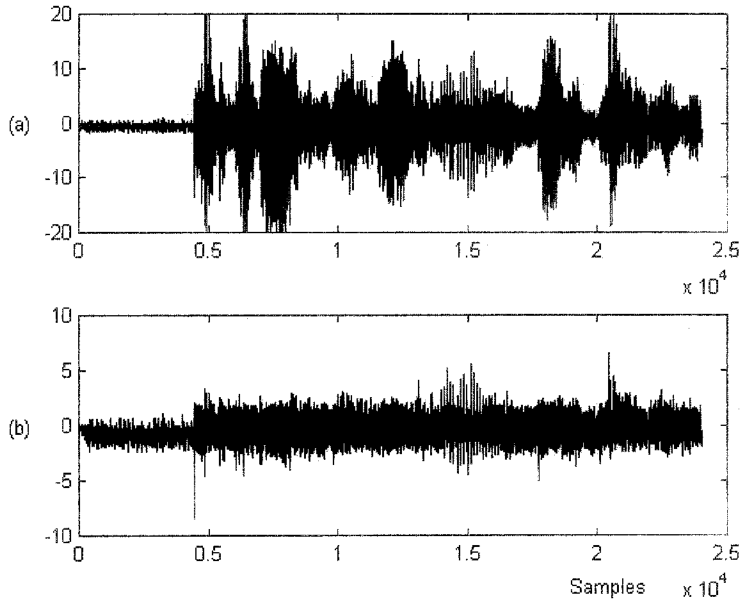

Figure 5. AVC output (bandwidth 10 Hz) for a speech waveform. (a) Original speech waveform $f_i(t)$, (b) AVC output $f_o(t)$.

AVC output, which has been scaled by the AVC so that the average variance is unity across the whole waveform.

For a bandwidth of 10 Hz, the AVC is then able to compress the speech so as to give unity variance at all times. However, the envelope of the speech is "flattened" for this high-bandwidth case, as is shown in Figure 5. Although the lower-bandwidth case would be used in practice, it is worth commenting that the intelligibility of the speech was unaffected by the AVC for this case.

The variance estimation loop was added to the AVC as in Figure 2 and was initially tested on a sinusoidal input. Figure 6a, b and c show the original 100 Hz sinusoid, the AVC output, and the variance estimator output, respectively. The variance converges to 0.5, which represents the rms value of the input signal squared. Tests were also carried out on other types of periodic waveforms.

For the unity white noise input waveform shown in Figure 7a, the variance estimation loop output (Figure 7c) converges to unity, and the AVC output (Figure 7b) remains unchanged.

Finally, if the bandwidth of the loops is taken as 10 Hz, then it is possible to track the variance of a speech signal with time. Figure 8a shows the original speech waveform, Figure 8b shows the AVC flattened output, and Figure 8c shows the variance estimator output. The AVC output has the "flattened" form discussed in the earlier example. However, this is of no consequence if it is the tracking ability of the variance that is primarily required, rather than setting the total average variance to a predefined setpoint.

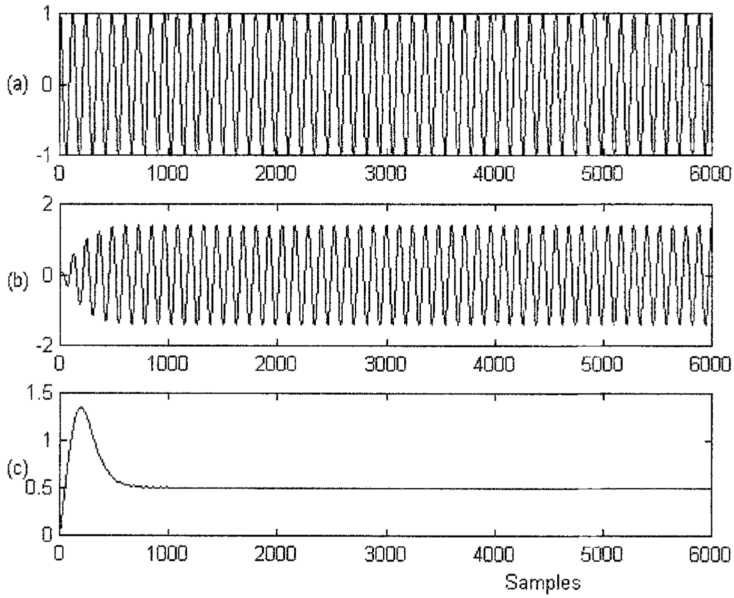

Figure 6. Variance estimation for a sinusoidal input. (a) Signal input to AVC $f_i(t)$, (b) AVC output $f_o(t)$, (c) variance estimator output $y_2(t)$.

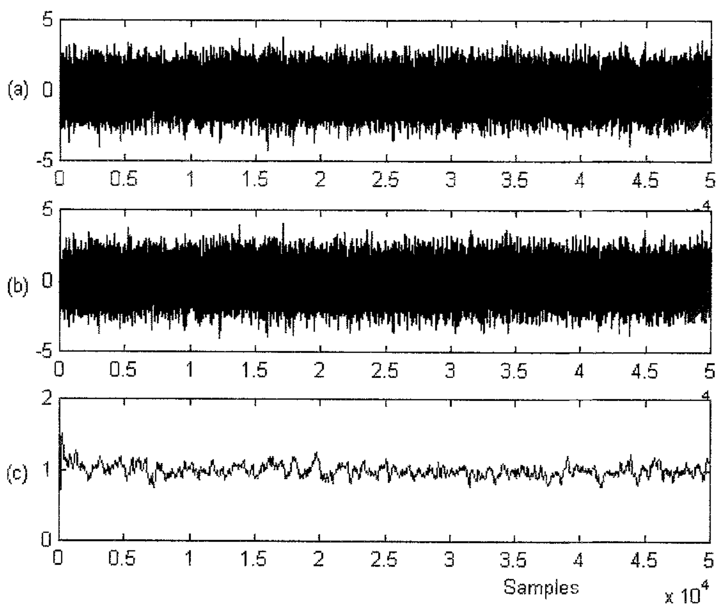

Figure 7. Variance estimation for white noise unit variance. (a) White noise input to AVC $f_i(t)$, (b) AVC output $f_o(t)$, (c) variance estimator output $y_2(t)$.

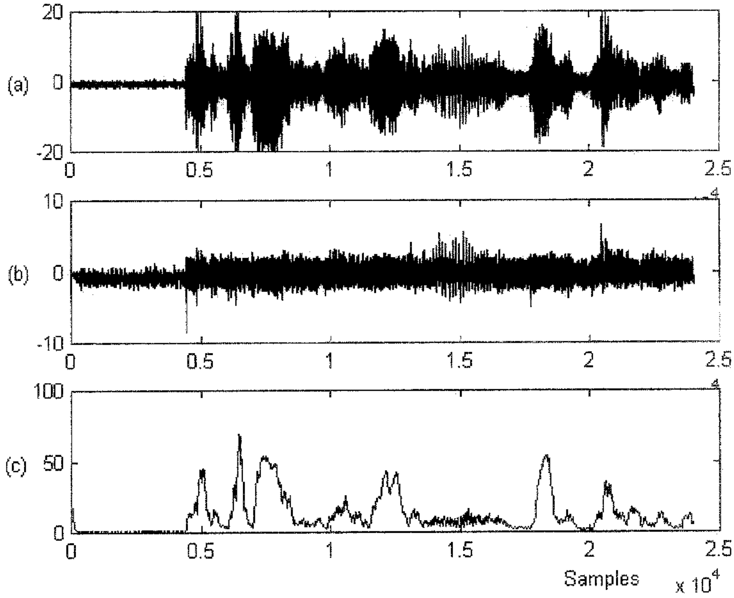

Figure 8. Tracking the variance of a speech waveform. (a) Speech waveform input to AVC $f_i(t)$, (b) AVC output $f_o(t)$, (c) variance estimator output $y_2(t)$.

6. Conclusions

A new closed-loop circuit termed *automatic variance control* has been presented which has the ability to alter the variance of a signal to a predefined value on a sample-by-sample basis rather than relying on batch-data type computations. A second control loop that interfaces to the AVC has the ability to estimate the variance of the original input signal and track it accurately with time. Several examples were shown to illustrate the operation of both loops. The new technique will have applications in areas of adaptive signal processing that are variance sensitive, such as LMS adaptive filters.

References

[1] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1986.

[2] T. J. Shan and T. Kailath, Adaptive algorithms with an automatic gain control feature, *IEEE Trans. Circuits and Systems*, 35, 1, 122–127, 1988.

[3] J. Tapia and S. M. Kuo, An optimized automatic gain controller for real time recursive LMS adaptive filters, in *Proc. 1990 IEEE International Symposium on Circuits and Systems*, Cat. No. 90CH2868-8, IEEE, 4, New York, NY, pp. 3154–3157, 1990.

[4] P. C. Young, *Recursive Estimation and Time Series Analysis: An Introduction*, Springer-Verlag, Berlin, 1984.