UNIVERSITY OF SOUTHAMPTON

institute of sound and vibration research

THE APPLICATION OF ADAPTIVE FILTERING TO THE ACTIVE CONTROL OF SOUND AND VIBRATION

by S.J. Elliott & P.A. Nelson

ISVR Technical Report No. 136

September 1985

THE APPLICATION OF ADAPTIVE FILTERING TO THE ACTIVE CONTROL OF SOUND AND VIBRATION

ISVR Technical Report No. 136

S.J. Elliott and P.A. Nelson

September, 1985

Authors: .....Group Chairman: .....

.....

CONTENTS

1. INTRODUCTION

There has recently been a surge of interest in 'active' methods of noise and vibration control, in which additional secondary sources are used to cancel the sound or vibration from the original 'primary' source. This interest has been spurred by recent developments in microprocessor technology, which make the economic implementation of such techniques increasingly feasible.

Although the most fundamental work in this area is probably that concerned with the physics of the interaction between the primary and secondary sources, there are also important questions to be answered regarding the methods by which the objectives of active control are achieved. Active control systems can normally only be formulated if it is assumed that superposition applies, i.e. that the system is linear, and that the properties of the primary field are constant i.e. the excitation is stationary. In many of the situations in which active noise control is applied, however, the properties of the sound or vibration field are mildly nonlinear and mildly nonstationary. An active control system must be adaptive in order to cope with these problems and consequently the use of adaptive methods in active control is widespread. The simplest form of such an adaptive system is one in which the 'best estimate' of a solution obtained assuming linearity and stationarity is tried, and after waiting for any transients to die away, the residual field is examined and the solution is adjusted to cancel this residue. Such 'block adaptive' schemes have been applied to the active control of random noise in ducts [1] and the multichannel control of harmonic vibration [2].

In the field of signal processing there is a class of adaptive system in which the properties of a digital filter are adjusted to minimise a single, electrical, error signal on a timescale which is much smaller than that used in the algorithms described above. There has been a considerable literature build up over the past two decades about such methods because of their importance in the fields of telecommunications, beamforming and bio-medical electronics. The continuing interest is also demonstrated by the fact that there have been at least four text books published in the last two years, specifically on the subject of adaptive filtering [3,4,5,6]. A very

similar technique has also been used over a similar period of time in control theory [13]. Unfortunately the similarity between the two traditions is not readily apparent from an examination of the two bodies of literature.

The application of this knowledge to the problem of active control is not, however, a straightforward matter. For example the use of an adaptive filter to drive a loudspeaker in an active noise control system which creates a 'zone of silence' in a room containing a noisy machine is, rather cryptically, left as an exercise for the reader in one recent book [ref. 6, page 300].

It is the purpose of this report to apply the philosophy adopted in the derivation of one widely used method for adaptive filter design, to the problem of the multidimensional active control of periodic excitations in physical systems.

The most general formulation of a multichannel active control system [8] involves the use of an array of sensors to detect the contributions from a number of primary sources before feeding them to an array of filters feeding the secondary sensors. The use of such detection sensors introduces the possibility of feedback paths from the secondary sources to these sensors. Thus the system becomes closed loop, even in the steady state, making the problem of adaptive adjustment difficult at best and, at worst, making the system unstable.

Under certain circumstances these feedback paths are eliminated or at least greatly reduced. A very important example of such a simplification occurs when the primary excitation is periodic and has a known fundamental frequency. Under these conditions a 'reference signal' can be generated with an identical fundamental frequency and containing all the harmonic components present in the primary field. This signal can now be used to drive the secondary sources, via a set of filters, thus dispensing with the detection sensors.

There are many important noise and vibration problems which are amenable to active control and also are nearly periodic. For example, the active control of sound and vibration from reciprocating or rotating machinery falls into this category. Two particular

problems which are currently being widely investigated are the active control of propeller induced cabin noise [9] and the 'higher harmonic' control of helicopter vibration [10, 11]. In practice both of these problems have been formulated in terms of the minimisation of the sum of the squared outputs from a number of error sensors (microphones or accelerometers) by adjusting the inputs to a number of sources (loudspeakers or shakers). These applications are thus examples of a more general class of multichannel or multivariate active control system with a primary field of known form.

This report presents the results of a preliminary investigation into the application of adaptive filtering to such multichannel active control systems. A review of the techniques used for single channel cancellation and their application to active control are first presented in Section 2. Section 3 introduces an algorithm which has been developed to extend such methods into multichannel applications, and discusses some of its properties. It is the authors' opinion that the careful application of adaptive filtering concepts to the problems of active control can be very fruitful. There are, however, a number of questions which remain unresolved and some of these form the basis of current work at ISVR.

2. ADAPTIVE FILTERING IN SINGLE CHANNEL SYSTEMS

2.1. ADAPTIVE NOISE CONTROL

Adaptive noise cancellation is the name used for a class of electrical adaptive systems typified by that shown in figure 2.1. It is assumed that a 'reference' signal is available which is perfectly correlated with the noise component of the 'primary' signal. This is passed through a filter and subtracted from the primary signal to form the 'error' signal. The canceller is adapted so that the filter output is as close a match to the noise component in the primary signal as can be obtained. The consequence of this is that the 'error' signal is a better estimate of the true 'signal', from the 'Signal Source'. Another result of this behaviour is that the mean square value of the error signal is minimised and this is the key to the operation of the canceller.

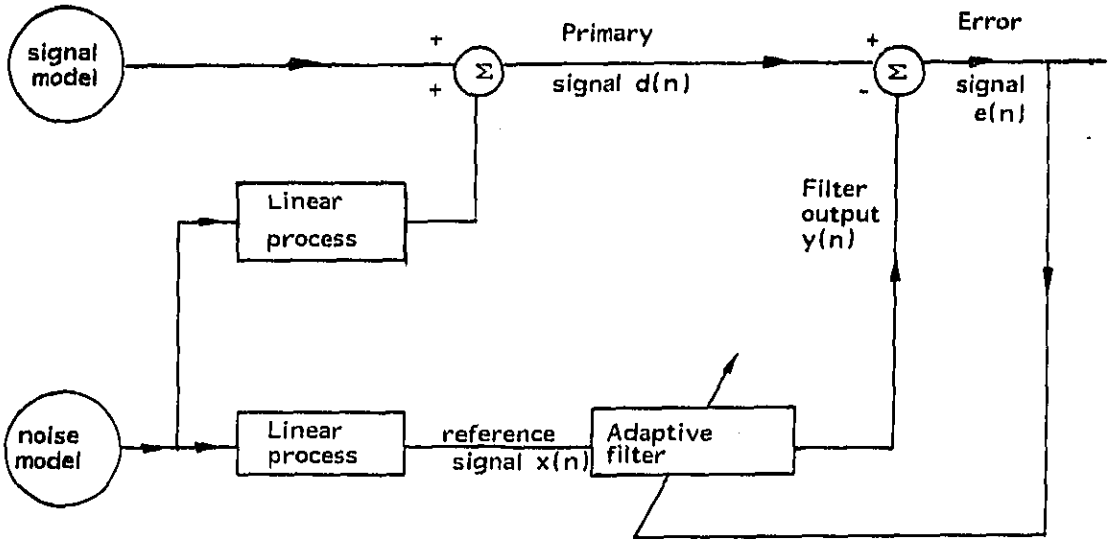

Figure 2.1. Block diagram illustrating a typical arrangement for electrical adaptive noise cancellation.

Assume that the system is sampled and that the filter is transversal (FIR), then:

$$y(n) = \sum_{i=0}^{I-1} w_i \, x(n - i) \qquad (2.1)$$

where w_i is the i'th coefficient of the filter. Thus,

$$e(n) = d(n) - \sum_{i=0}^{I-1} w_i \, x(n - i) \qquad (2.2)$$

Since e(n) is a linear function of each filter coefficient, the mean square value of e(n), which is equal to the total error E, must be a quadratic function of each w_i. The mean square value of e(n) can never be negative, so the I + 1 dimensional error surface formed by plotting the total error against each of the I filter coefficients must have a unique global minimum. Due to the quadratic nature of this error surface simple gradient descent methods can be used to adapt the coefficients in order to minimise the total error.

One such technique is the method of steepest descent where each coefficient is adjusted at each iteration by an amount proportional to the negative value of the gradient of the total error with respect to that coefficient, i.e.:

$$w_i(k+1) = w_i(k) - \mu \frac{\partial E}{\partial w_i} \qquad (2.3)$$

for the k'th iteration.

Since $E = \overline{e(n)^2}$, where the bar indicates a time average,

$$\frac{\partial E}{\partial w_i} = \frac{\partial \overline{e^2}}{\partial w_i} = 2 \, \overline{e(n) \frac{\partial e(n)}{\partial w_i}} \qquad (2.4)$$

but from equation 2.2.

$$\frac{\partial e(n)}{\partial w_i} = -x(n-i) \qquad (2.5)$$

The steepest descent algorithm thus becomes

$$w_i(k+1) = w_i(k) + 2\mu \ \overline{e(n)\ x(n-i)} \qquad (2.6)$$

If the instantaneous gradient is used to update the coefficients and all the coefficients are adjusted at every sample point then the algorithm becomes:

$$w_i(n+1) = w_i(n) + \alpha\ e(n)\ x(n-i) \qquad (2.7)$$

where $\alpha = 2\mu$. This is known as the LMS algorithm in the signal processing literature and is attributed to Widrow and Hoff [12]. It is also a special class of Stochastic gradient algorithm discussed in the control literature [13]. Although the detailed origins of such stochastic gradient algorithms appear rather obscure, they have obviously grown out of methods of numerical analysis used since the 19th Century.

This stochastic gradient (or SG) algorithm has been found to converge under a wide variety of conditions and is very straightforward to program for real time digital signal processing. Consequently it has become very popular in a wide variety of applications [6]. The algorithm has been widely analysed in terms of its stability, convergence time, and misadjustment error (due to the coefficients continuously being changed even when the bottom of the error surface has been reached). Most theoretical analysis make an implicit assumption that the filter is converging slowly. This is often expressed in the statement that the instantaneous values of the coefficients are independent of the current input signal. Theoretical treatments which deal with the full dynamic behaviour of the algorithm have only recently begun to emerge [5 (page 48) 14]. A brief discussion of the most widespread current theoretical treatment is, however, helpful in understanding some of the limitations of such algorithms.

The discussion below is based on that presented by Widrow in 1971 [15] and used in the book by Widrow and Stearns [6]. The analysis is based on expressing the update equation 2.7 in matrix form for all coefficients.

Let

$$\underline{W}_n = \left[w_0(n)\ w_1(n)\ w_2(n)\ \ldots\ w_{I-1}(n)\ \right]^T$$

$$\underline{X}_n = \left[x(n)\ x(n-1)\ x(n-2)\ \ldots\ x(n-I+1)\right]^T$$

Equation 2.7 can now be expressed as

$$\underline{W}_{n+1} = \underline{W}_n + \alpha\, e(n)\, \underline{X}_n \qquad (2.8)$$

the output of the filter can also be conveniently expressed as

$$y(n) = \underline{X}_n^T\, \underline{W}_n, \quad \text{so} \quad e(n) = d(n) - \underline{X}_n^T\, \underline{W}_n \qquad (2.9)$$

Thus,

$$\underline{W}_{n+1} = \underline{W}_n + \alpha(d(n)\, \underline{X}_n - \underline{X}_n\, \underline{X}_n^T\, \underline{W}_n) \qquad (2.10)$$

The analysis at this point is completely general but to make the algebra tractable, the assumptions of a stationary random input and of 'slow' adaption are used. The stationarity of the input is first used to define an estimation value of all quantities for a large number of input sequences generated by the same stochastic process. Secondly we assume that the instantaneous value of the coefficients expressed in the 'weight vector' $\underline{W}_n$ are uncorrelated with the current set of inputs in $\underline{X}_n$. This second assumption is obviously dubious for fast adaption rates [14] but for 'slow' adaption it does not appear unreasonable, and does lead to a relatively straightforward average update equation. The asumption allows expectation values of $\underline{X}_n\, \underline{X}_n^T$ and $\underline{W}_n$ to be evaluated separately, and so the average behaviour of the coefficients, over an ensemble of inputs, can be expressed as:

$$\underline{W}_{n+1} = \underline{W}_n + \alpha\, (\underline{P} - \underline{R}\, \underline{W}_n) \qquad (2.11)$$

where $\underline{P} = E(d(n)\, \underline{X}_n)$ and $\underline{R} = E(\underline{X}_n\, \underline{X}_n^T)$

i.e. $\underline{P}$ is the vector of cross correlations and $\underline{R}$ is the matrix of input autocorrelations, E is the expectation operator.
The term in brackets in equation 2.11 corresponds to the average gradient, and by setting this to zero for all coefficients, we obtain the optimum, or Wiener, set of coefficients:

$$\underline{W}_{opt} = \underline{R}^{-1} \underline{P} \qquad (2.12)$$

By expressing $\underline{P}$ in terms of $\underline{R}$ and $\underline{W}_{opt}$ using this expression, and defining

$$\underline{V}_n = \underline{W}_n - \underline{W}_{opt} \qquad (2.13)$$

as the vector of coefficient misadjustment, one obtains the equation

$$\underline{V}_{n+1} = \underline{V}_n - \alpha \underline{R} \underline{V}_n = [\underline{I} - \alpha \underline{R}] \underline{V}_n \qquad (2.14)$$

Thus, by induction,

$$\underline{V}_n = [\underline{I} - \alpha \underline{R}]^n \underline{V}_o \qquad (2.15)$$

The convergence of the algorithm indicated by the vector $\underline{V}_n$ tending to zero as n becomes large and so is dependent on the values of $\underline{R}$. This dependence can be made clearer if the misadjustment vector is transformed from $\underline{V}$ to $\underline{V}'$, where:

$$\underline{V}' = \underline{Q}^{-1} \underline{V} \quad , \quad \therefore \underline{V} = \underline{Q} \underline{V}' \qquad (2.16)$$

and $\underline{Q}$ is the matrix of normalised eigenvectors of $\underline{R}$, with the property that $\underline{Q}^{-1} \underline{Q} = \underline{I}$.

This transformation eliminates the cross-coupling between the actions of the various filter coefficients due to the off-diagonal terms of $\underline{R}$ in equation 2.15. This can be demonstrated by writing

$$\underline{R} = \underline{Q} \underline{\Lambda} \underline{Q}^{-1} \qquad (2.17)$$

where $\underline{\Lambda}$ is a diagonal matrix containing the eigenvalues of $\underline{R}$.

By using these expressions for $\underline{V}$ and $\underline{F}$ (equations 2.16 and 2.17) in equation 2.14 we obtain:

$$\underline{V}'_{n+1} = \underline{V}'_{n} - \alpha \underline{\Lambda} \underline{V}_{n}' \tag{2.18}$$

The second term on the R.H.S. of the equation represents $(-\alpha)$ multiplied by the gradient of the error with respect to V_n'. Since $\underline{\Lambda}$ only has diagonal terms, it is clear that the gradient of the error with respect to each coefficient of $\underline{V}_n'$ can only be depenent on that value. The set of coefficients in $\underline{V}'$ must thus describe the principal axes of the error surface. Rewriting equation 2.18 as

$$\underline{V}'_{n+1} = (\underline{I} - \alpha \underline{\Lambda}) \underline{V}_{n}' \tag{2.19}$$

the time history of the ith coefficient of $\underline{V}'_n$, $V_i'(n)$ say, can now be written as a scalar equation,

$$V_i'(n+1) = (1 - \alpha\lambda_i) V_i'(n) \tag{2.20}$$

since equation 2.19 describes a set of independent difference equations, λ_i is thus the ith eigenvalue of $\underline{R}$.

This set of equations has been used to describe both the stability, and the speed of convergence of the algorithm. Equation 2.20 describes a first order recursion whose transient response is a single exponential. In order for the algorithm to be stable the magnitude of $V_i'(n+1)$ must be less than that of $V_i'(n)$ thus

$$|1 - \alpha \lambda_i| < 1 \tag{2.21}$$

$$\therefore \quad 1 > 1 - \alpha \lambda_i > -1 \quad \text{or} \quad 0 < \alpha < \frac{2}{\lambda_i} \tag{2.22}$$

Since each λ_i is real and positive. This condition will be most stringent for the largest eigenvalue, λ_{max}, which will determine the largest value of α which can be used while keeping the algorithm stable. i.e.

$$\alpha < \frac{2}{\lambda_{max}} \tag{2.23}$$

However, λ_{max} is less than or equal to the trace of the autocorrelation matrix, which is approximately equal to the mean squared value of the reference signal multiplied by the order of the filter (I). So

$$\lambda_{max} < I \ \overline{x^2} \qquad (2.24)$$

and a condition for stability which is more restrictive than 2.23 but more easily applied is that

$$0 < \alpha < \frac{2}{I \ \overline{x^2}} \qquad (2.25)$$

The time constant of convergence of $V_i'(n)$, defined as the number of samples taken for it to fall by $\frac{1}{e}$, may be determined for equation 2.20 as

$$\tau_i = \ln (1 - \alpha\lambda_i) \simeq \frac{1}{\alpha\lambda_i} \qquad \text{if } \alpha \lambda_i << 1 \qquad (2.26)$$

The equation which takes longest to converge, i.e. has the largest time constant, τ_{max}, will be that containing the smallest values of λ_i, λ_{min} say,

$$\tau_{max} \simeq \frac{1}{\alpha \lambda_{min}} \qquad (2.27)$$

This term will generally dominate the convergence of the filter, so we let $\tau_{max} = \tau$, an estimate of the overall time constant of convergence. To obtain the smallest value of this overall time constant we let α take its largest value consistent with stability given by equation 2.23, to obtain:

$$\tau > \frac{\lambda_{max}}{2 \ \lambda_{min}} \qquad (2.28)$$

This is a well known result and is widely quoted as being the major drawback of the SG algorithm when applied to signals with a large 'eigenvalue spread'. A practical estimate of the ratio of the

maximum to minimum eigenvalue (sometimes called the condition number of the matrix [16]) can be obtained by using the fact that this ratio is always less than or equal to the square of the modulus of the largest spectral component of the reference signal divided by the square of the modulus of its smallest spectral component [16]. Thus the eigenvalue spread is related to the dynamic range of the spectrum of the reference signal.

One area in which adaptive filters are widely used is in communications channels. The signals in such systems are often speech waveforms, whose spectrum has a large dynamic range. The performance of standard SG algorithms in this application has thus been disappointing because of the large eigenvalue spread in the signals. This has prompted the search for modified algorithms which avoid this convergence time problem. A good review of one such method is given in [16], in which the input signal is transformed before being filtered in an attempt to ensure that the adaptive coefficients used on the transformed data correspond to the principal axis of the error surface. Although the transformation which optionally achieves this objective is rather obscure (the Karlunen-Loeve transformation), it is shown that the Discrete Fourier Transformation has a nearly-optimal performance. This further encourages an interpretation of the frequency domain as being a useful way of viewing the eigenvalues which are, after all, the strengths of the different orthogonal components in the reference signal. The frequency domain LMS algorithm was originally introduced [17] as an efficient implementation of the time domain LMS algorithm. It has been found, however, that by choosing the convergence coefficient in each frequency bin to be inversely proportional to the signal power in that bin, considerable improvements in adaption time are achieved. In general terms, this corresponds to being able to choose α separately for each of the decoupled recursive equations 2.20 ensuring that each of the 'modes' decays at the same rate.

Other approaches to adaptive filtering which are potentially faster than the time domain LMS algorithm are those using lattice structures [ref 6, p 164] and those using recursive least squares or fast Kalman algorithms [18].

2.2. ADAPTIVE CANCELLATION OF PERIODIC SIGNALS

If the reference signal in an adaptive canceller is made up of a number of sinusoids, considerable simplifications can occur in both the implementation and analysis of the algorithm. The latter was originaly investigated by Glover [19] who showed that the behaviour of such an adaptive canceller could be approximated by a linear time invariant filter. The approximations become exact if the sinusoids in the reference signal have an exactly integer number of samples per cycle, in other words they are synchronously sampled, and the filter length contains an integer number of cycles of the reference signal. Under this condition the behaviour of the system, from the primary signal to the error signal, can be described exactly by that of a linear time invariant notch filter, the bandwidth of which is determined by the convergence coefficient.

Only two filter coefficients are needed for each sinusoid in the reference signal, which can lead to considerable computational savings.

To illustrate these points a simple simulation of an adaptive canceller is presented with a single sinusoidal reference signal, sampled at exactly four samples per cycle. The normalised frequency of the reference signal is that exactly $^1/_4$ and

$$x(n) = \cos(n\pi/2) \qquad (2.30)$$

This signal may be generated using the recursion equation:

$$x(n) = -x(n-2) \quad \text{with} \quad x(0) = 1, \quad x(-1) = 0 \qquad (2.31)$$

One advantage of using such a reference signal is that only a two point FIR filter need be used to generate signals with arbitrary amplitude and phase, and this filtering operation has the interpretation that if

$$y(n) = w_0\, x(n) + w_1\, x(n-1) \qquad (2.32)$$

this is equal to

$$y(n) = w_0 \cos(\pi n/2) + w_1 \sin(n\pi/2) \qquad (2.33)$$

In this case, there is no distinction between a time domain algorithm and a frequency domain algorithm since the two coefficients are already operating on the 'real' and 'imaginary' parts of the only frequency present.

The primary signal in the simulation has the form

$$d(n) = -1.0 \cos(n\pi/2) + 0.9 \sin(n\pi/2) \qquad (2.34)$$

A two coefficient filter is updated using the standard SG algorithm to minimise the difference between its output, y(n), and the primary signal d(n). The instantaneous value of this error singal e(n) is shown plotted for 128 samples in figure 2.2. Also shown is the square of this signal averaged with a two point moving average filter,

$$E(n) = (e^2(n) + e^2(n-1))/2 \qquad (2.35)$$

and the trajectories of the two coefficients. Since e(n) has a normalised frequency of $^1/_4$, the alternating part of $e^2(n)$ has a normalised frequency of $^1/_2$ and may be accurately averaged over two points, as in equation 2.35.

A convergence coefficient of $\alpha = 0.1$ was used in this simulation, although values of up to $\alpha = 2$ were stable, and the fastest convergence is given when $\alpha = 1$ in which case the filter adapts within one cycle.

If the standard theory, outlined above, is used to analyse this system, the task is considerably simplified by noting that for $x(n) = \cos(n\pi/2)$,

$$\underline{R} = {}^1/_2 \, \underline{I} \qquad (2.36)$$

and the two eigenvalues of $\underline{R}$ are both equal to $^1/_2$.

The convergence of the coefficients is thus already independent, without transformation. The update equation 2.14 thus becomes

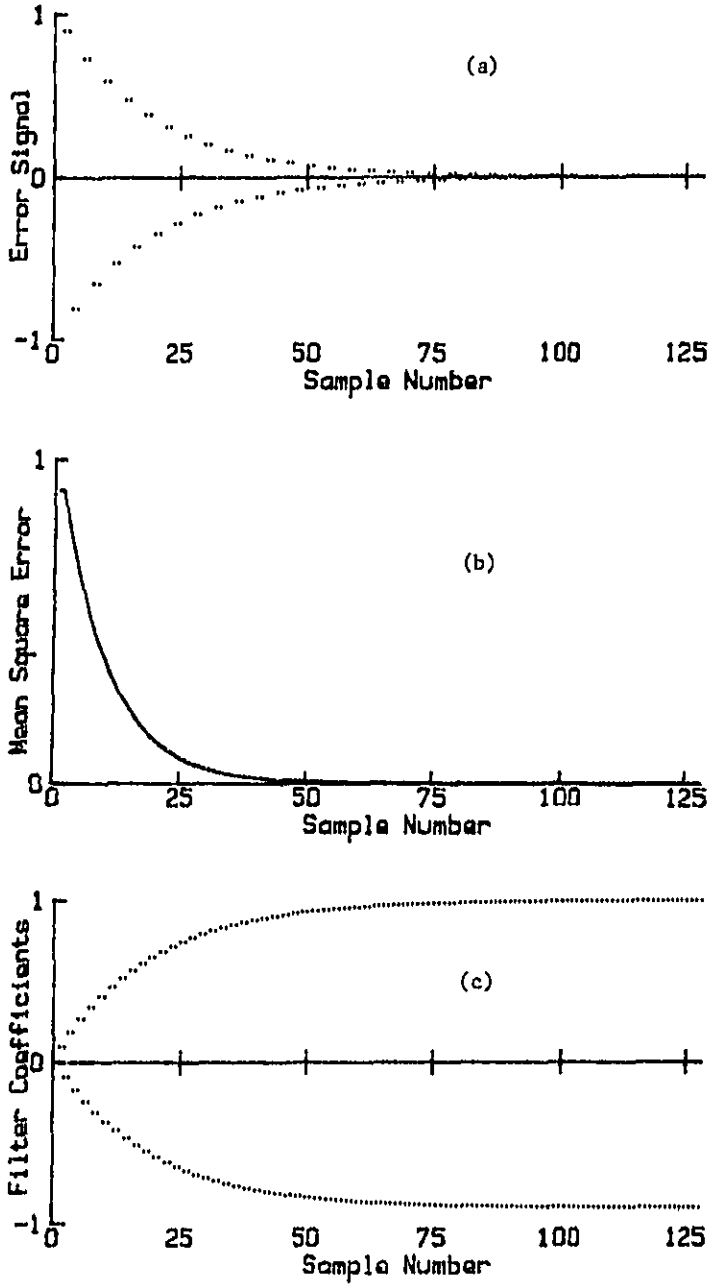

Figure 2.2. The behaviour of a synchronously sampled adaptive noise canceller with $\alpha = 0.1$;
a) instantaneous error signal, b) averaged error,
c) trajectories of coefficients.

$$\underline{V}_{n+1} = (1 - \alpha/2)\,\underline{V}_n \qquad (2.37)$$

Not only is the convergence of the coefficients independent according to this equation, but their time constants of convergence are also predicted to be equal.

This is exactly as observed in figure 2.2., although the predicted time constant of convergence is found to be in error for fast adaption. Equation 2.33 predicts that if $\alpha = 2$, so that the term in brackets is zero, then the system will adapt within one cycle. It also predicts that the system will be stable if the term in brackets is nearly equal to -1 i.e. α is nearly equal to 4. This is clearly at variance with what is found in the simulation, and the error comes from the assumption of 'slow' adaption in the derivation in Section 2.1.

Applying Glovers analysis [19] to this system, the equivalent transfer function between the error signal and desired signal is found to be equal to:

$$\frac{E(Z)}{D(Z)} = \frac{Z^2 + 1}{Z^2 + 1 - \alpha} \qquad (2.38)$$

This transfer function has a pair of zeros at $Z = \pm j$ and a pair of poles at $Z = \pm j\sqrt{\alpha - 1}$. If $\alpha = 0$ the poles and zeros are coincident and the transfer function is unity, i.e. it never adapts, as expected. If $\alpha = 1$ the poles will lie at the origin of the Z plane, and the transfer function loses its recursive form, becoming an entirely FIR system and having a response that dies out within two samples. If α approaches 2 the position of the poles approach ± 1, which predicts that the system is unstable for $\alpha = 2$. All of these properties are found exactly in the simulation which suggests that the equivalent transfer function approach does predict the full 'dynamic' behaviour of the algorithm.

The difference equation between the error sequence and the desired sequence, corresponding to the transfer function in equation 2.38, is

$$e(n) = d(n) + d(n-2) - (1 - \alpha)\, e(n-2) \qquad (2.39)$$

Note that if the desired signal has the same frequency as the reference it will also obey the difference equation $d(n-2) = -d(n)$, by analogy with equation 2.31, and the difference equation becomes

$$e(n) = (\alpha - 1)\, e(n-2) \qquad (2.40)$$

This is similar in form to equation 2.37, and predicts the error response to be that of a simple first order system (at half the sample rate of $d(n)$) as found in the simulations.

If a number of harmonic frequencies are present in the primary input it has been shown [20] that there are considerable advantages in using a reference signal which is a periodic inpulse train with the same fundamental period as the primary, say N samples, so

$$x(n) = \sum_{k=-\infty}^{\infty} \delta(n - kN) \qquad (2.41)$$

This can reduce the number of operations required to implement the algorithm from 2N multiplications and 2N additions per sample to one addition per sample. This reference signal also leads to a particularly simple equivalent transfer function.

The behaviour of this harmonic algorithm can also be analysed using the method outlined in Section 2.1. Although we cannot expect to obtain reliable predictions of the response if the system is adapting 'quickly', some additional insights are gained by examining the autocorrelation matrix which has the special form in this case; $\underline{R} = \underline{I}$. So again the convergence of the coefficients is independent and they all converge with the same time constant since the eigenvalues are equal. If a reference signal of this form can be used the convergence of the algorithm will be as fast as any other algorithm which attempts an orthogonalisation of the input prior to adaption.

2.3. APPLICATION TO ACTIVE CONTROL

An active control system with a single secondary source and a single error sensor can be considered as a generalisation of an adaptive canceller. The most important difference is the introduction between the filter output and error summing junction of an extra transfer function to account for the responses of the source and sensor and that of the system to be controlled. This secondary path, with transfer function C, is illustrated in figure 2.3 in which the notation normally associated with active noise control has been substituted for the signal processing nomenclature used in figure 2.1. Note that s(n) is now added to d(n), rather than being subtracted as in figure 2.1. This has no effect on the analysis leading to the filter update equation above, equation 2.6, except to change the sign of the update term.

The introduction of the secondary path transfer function into a canceller using the standard SG algorithm above will generally cause instability [21]. This is because the error signal is not correctly 'aligned' in time with the reference signal, due to the presence of C, when the gradient estimate is calculated.

There are a number of possible schemes for realigning these signals so as to obtain an unbiased gradient estimate and thus achieve convergence. One method has been recently termed the 'filtered x' algorithm [reference 6, p 288], although it was also reported by Burgess in 1981 [22], and has an intuitive interpretation in terms of the block diagram of figure 2.3. If the adaptive filter is assumed to be changing only very slowly compared to the timescale of the dynamic behaviour of the error path C, then the system is nearly equivalent to that drawn in figure 2.4, with these elements reversed. If we now redefine the reference signal as being r(n) in figure 2.4., it is clear that a standard SG algorithm could be used on this equivalent system. The update equation in this case would be:

$$w_i\,(n+1) \;=\; w_i(n) \;-\; \alpha\, e(n)\, r(n-i) \qquad (2.42)$$

However the signal r(n) is not available in the physical system of figure 2.3. An approximation to it can, however, be artificially generated by passing x(n) through an electrical filter which is

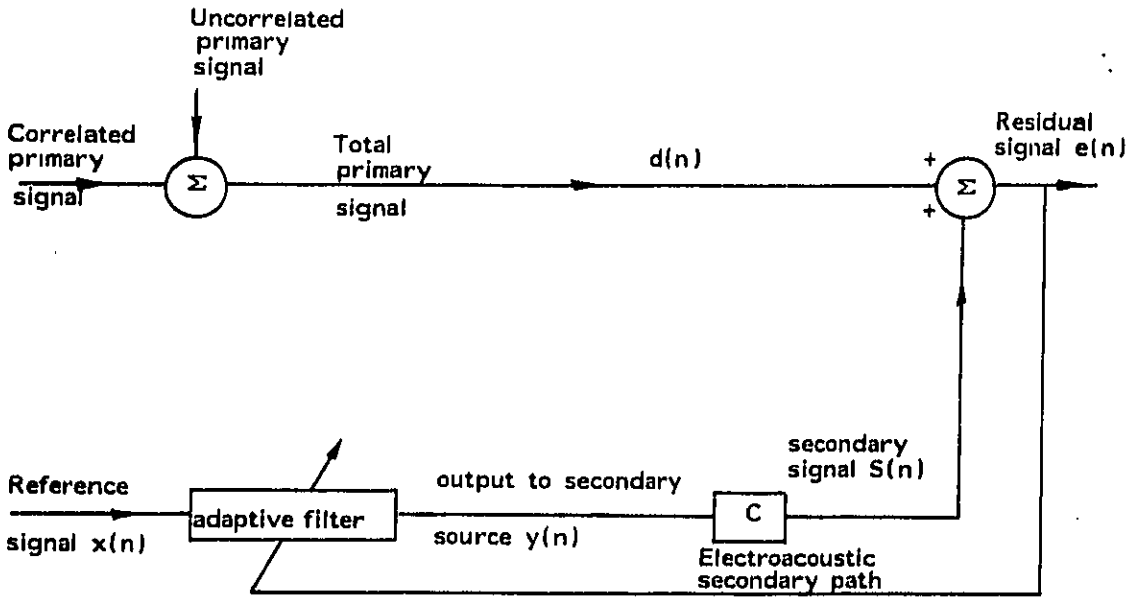

Figure 2.3. Block diagram illustrating single channel active noise control in the form of an adaptive noise canceller.

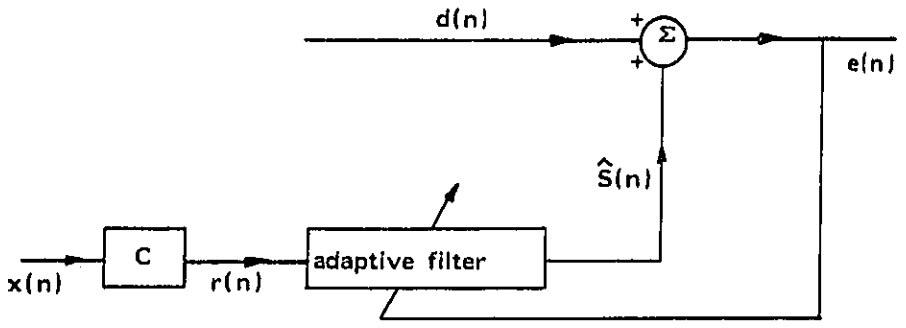

Figure 2.4. Block diagram to explain the filterd x algorithm.

arranged to have a similar response to that of the error path C.

When this algorithm is implemented two suprising things are observed. Firstly the convergence of the filter can be achieved much more quickly than the above argument would suggest, and secondly the algorithm appears to be very tolerant of errors made in the estimation of C by the filter generating r(n).

2.4. FILTERED x ALGORITHM AT A SINGLE FREQUENCY

It has been found that considerable insight can be obtained into the behaviour of the filtered x algorithm if a single, synchronously sampled sinusoid is used as a reference signal. An exact theoretical analysis of the algorithm can be performed in this case, which is presented in the next section. We concentrate here on the results of a computer simulation using a reference singal $x(n) = \cos(n\pi/2)$, as in section 2.2. above.

The block diagram of the simulation is shown in figure 2.5. in which the primary signal is derived from the reference with a 2 point FIR filter, and the error path is modelled as a pure delay of 10 samples. The signal r(n) is generated by passing x(n) through an exact version of the error path, although in this case delays of integer periods of the reference have no effect and a delay of two samples produces the same signal as a delay of ten samples.

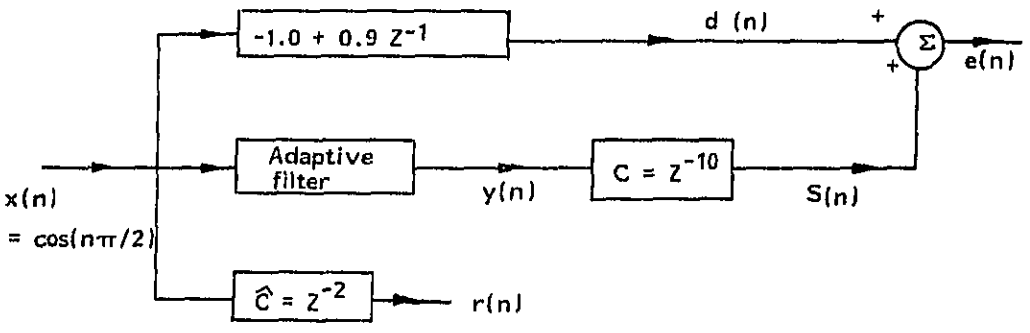

Figure 2.5. Block diagram of the simulation performed to investigate the performance of the filtered x algorithm with a sinusoidal reference signal.

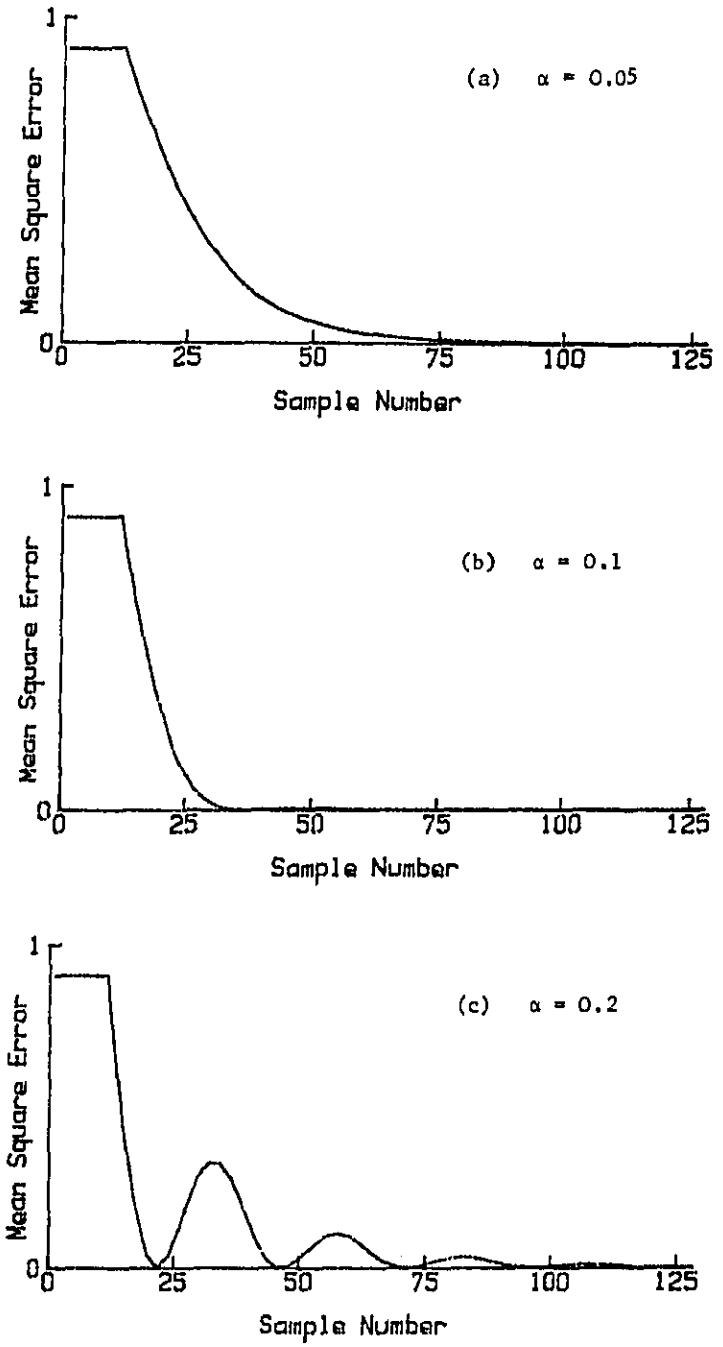

Figure 2.6. The mean square error signal produced by the simulation for a) α = 0.05, b) α = 0.1, c) α = 0.2.

Figure 2.6 shows the square of the error signal, averaging over two samples using equation 2.35, for 128 samples of the simulation using various values of the convergence coefficient, α. It is found that values of α approaching unity do not give a stable system, in contrast to the case where the error path is absent as considered above. The fastest convergence is observed with $\alpha \approx 0.1$, as in figure 2.6b. It is instructive to also observe the behaviour of the averaged error signal for various other values of α. For example those for $\alpha = 0.05$ and $\alpha = 0.2$ are presented in figures 2.6a and c. It is striking how these graphs are reminiscent of the overdamped and underdamped response of a simple second order system. Figure 2.6b would, by analogy, correspond to the case of critical damping. Although the error for all cases presented in figure 2.6 eventually decays away to zero, unstable behaviour is observed when α is greater than about 0.3, in which case the amplitude of the oscillations observed in figure 2.6c grow with each cycle. If $\alpha \approx 0.3$, the period of these oscillations corresponds to the delay in the error path, if α is reduced the period of oscillations becomes progressively longer as they become more heavily damped.

The convergence behaviour of this system with various other delays in the error path has also been investigated for various values of α. The 'Convergence time' of adaption, measured as being when the averaged error decays below 5% of its initial value and does not subsequently rise above this value, is plotted against α for these simulations in figure 2.7. The different curves correspond to various values of pure delay in the error path. The convergence time of the algorithm with no error path is given by the diagonal line and it is interesting to note how all the curves tend to this one for very small values of α. With no error path the convergence time is inversely proportional to α, in agreement with equation 2.26. The value of α for which the convergence time is fastest is plotted against the delay in the secondary path in figure 2.8. The optimum value of α is found to be approximately equal to $1/4D$ in this simulation, where D is the number of cycles delay in the secondary path. This gives a quantitative form to the intuitive result that the convergence must slow down as the response of the secondary path gets longer.

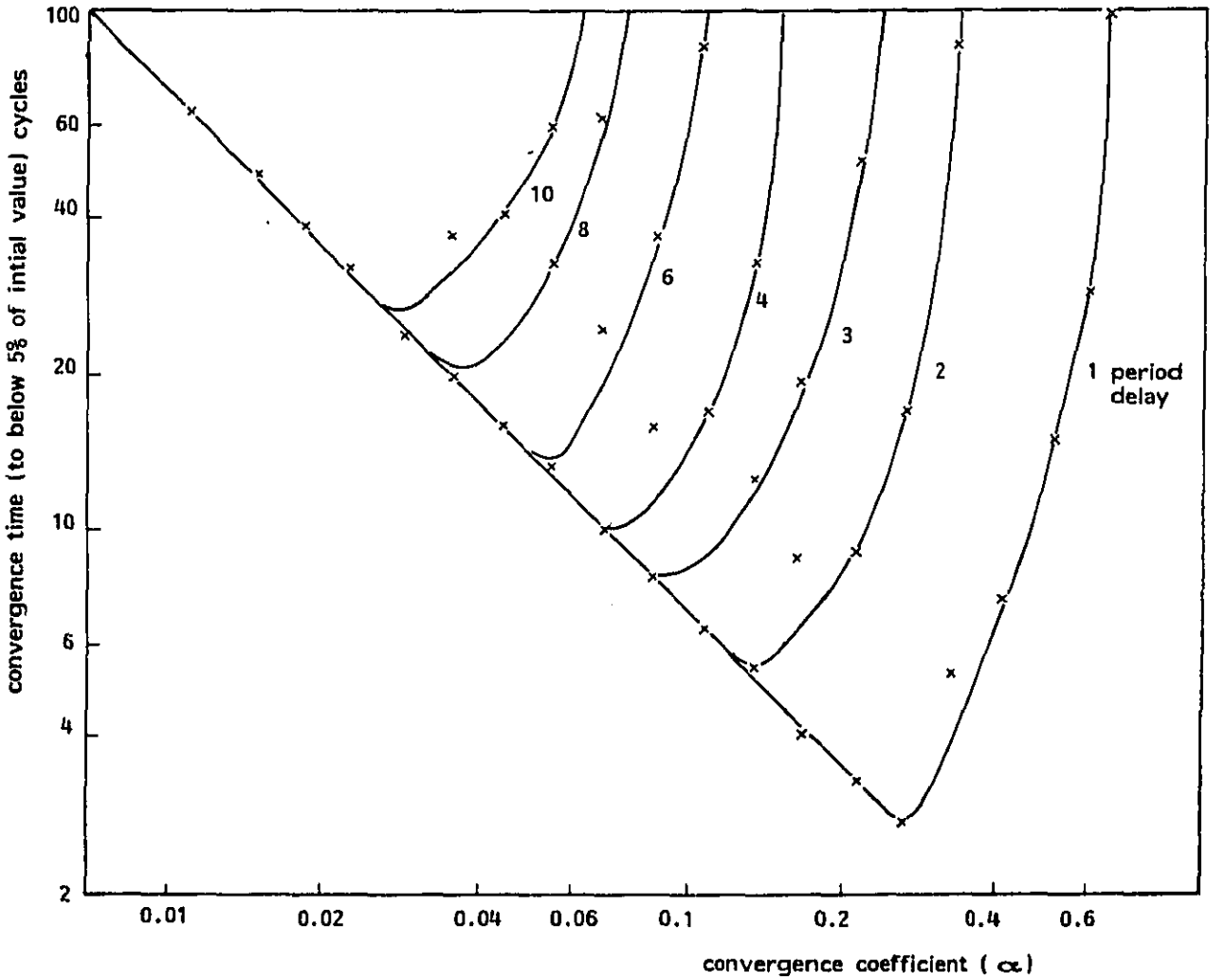

Figure 2.7. Graph of the 'convergence time', defined as being when the averaged squared error decays below 5% of its initial value, against the convergence coefficient for various values of pure delay in the error path.

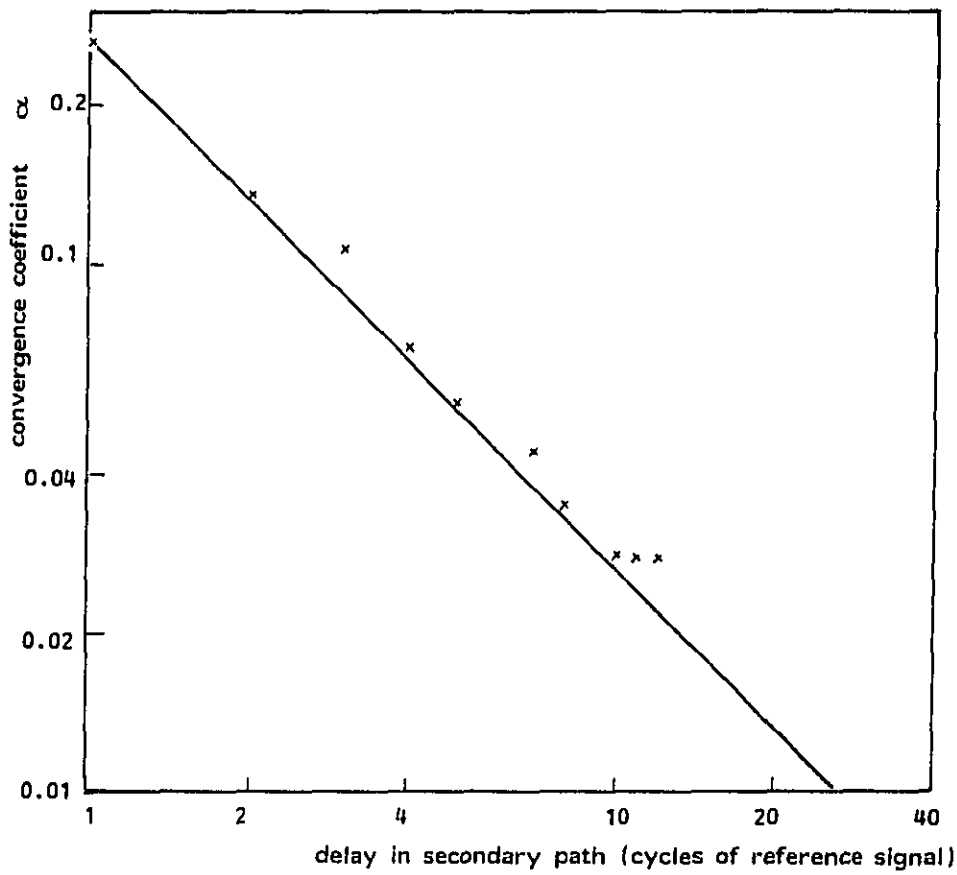

Figure 2.8 *Graph of the log of the convergence coefficient which gives the fastest convergence time, against the log of the delay in the secondary path.*

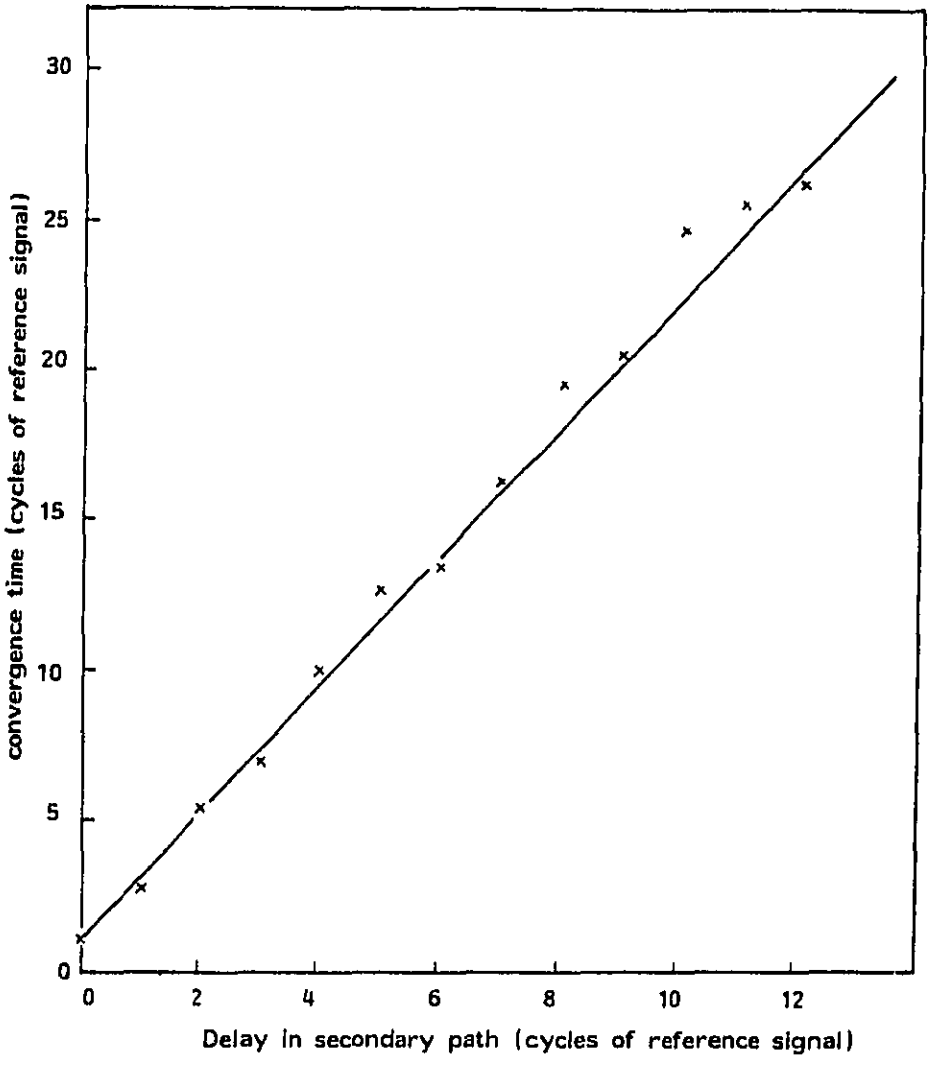

Figure 2.9. Graph of the minimum convergence time of the algorithm against the delay in the error path.

Figure 2.7 can also be used to obtain the minimum convergence time for a variety of delays in the error path. This is plotted in figure 2.9, which illustrates that the minimum convergence time rises in proportion to the delay in the secondary path. In this simple case, the minimum time for the total mean square error to converge to within 5% of the final value is about twice the delay in the secondary path. If the log of the mean square error is plotted against time in the simulations, two distinct regions of the convergence are observed. Firstly there is a period during which the mean square error does not change. This lasts for a time approximately equal to the delay in the secondary path and is obviously due to the effects of the adaptive filter taking some time to propagate through to the error. Secondly the decay of the error has an envelope which is approximately exponential, whether the decay is monotonic or oscillatory. In the simulations performed above the sum of this initial delay and time taken for the exponential decay to 5% of the initial mean square value (i.e., an attenuation of 13 dB) was found to be approximately twice the secondary delay. Thus the fastest exponential decay rate is about 13 dB per unit of secondary delay, and the time constant for this exponential decay to fall to 1/e of its initial value (i.e., -8.7 dB), is approximately 0.67 times the secondary delay.

These observations can be used to predict the convergence time defined according to different criterion to that used above. For example, the time for the mean square error to fall to 1% of its initial value (-20 dB) will be approximately (1 + 0.67 × 20/8.7) = 2.5 times the secondary delay. Similar trends in convergence time are observed in simulations with more complicated secondary paths, in which recursive terms as well as an overall delay are used.

2.5. ANALYSIS OF SINGLE FREQUENCY ALGORITHM

In this section we use the philosophy of approach developed by Glover [19] to obtain an equivalent transfer function between the desired signal, considered as the input to the system, and the error signal, considered as the output of the system. It is found that if the

reference is a synchronously sampled sinusoid, or the sum of a number of these, then the adaptive canceller behaves exactly like a linear, time invariant system between this input and output. The transfer function can thus be used to calculate the response of the system to any input excitation and also, more importantly, can be used to investigate the stability of the algorithm by examining the positions of the poles of the transfer function.

Using the nomenclature in figure 2.5, in which the adaptive filter is fed from a reference signal of the form

$$x(n) = \cos(\omega_o n) \tag{2.43}$$

The output of which passes through a secondary path filter (C) before being added to a desired signal to form an error signal, which is fed back to the adaptive filter. A filtered x algorithm is used to adapt this filter, so that

$$w_i(n+1) = w_i(n) - \alpha\, e(n)\, r(n-i) \tag{2.44}$$

The filtered reference signal, r(n) is formed by passing x(n) through an estimate ($\hat{C}$) of the true secondary path. By making this filter different from C the effect of errors in the estimate of the error path transfer function can be investigated.

The filter $\hat{C}$ is, however, only excited by x(n) at the reference frequency, ω_o. So if the modulus and phase of its transfer function at this frequency are;

$$\hat{C}(e^{j\omega_o}) = Me^{j\Phi}$$

the filtered reference signal must be

$$r(n) = M\cos(\omega_o n + \Phi) \tag{2.45}$$

$$\therefore \quad r(n-i) = M\cos(\omega_o n + \Phi - \omega_o i)$$

$$= M/2\left[e^{j\omega_o n}\, e^{j(\Phi - \omega_o i)} + e^{-j\omega_o n}\, e^{-j(\Phi - \omega_o i)}\right] \tag{2.46}$$

In the update term for the coefficient w_i, this signal is multiplied by e(n). If the Z transform of e(n) is E(z), the Z transform of the product e(n) r(n-i) is

$$Z\left\{e(n)\ r(n-i)\right\} = \frac{M}{2}\left[e^{j(\Phi-\omega_o i)}\ E(z\ e^{-j\omega_o}) + e^{-j(\Phi-\omega_o i)}\ E(z\ e^{j\omega_o})\right] \quad (2.47)$$

Now taking the Z transform of the update equation, with $W_i(z)$ as the Z transform of $w_i(n)$:

$$z\ W_i(z) = W_i(z) - \frac{\alpha M}{2}\left[e^{j(\Phi - \omega_o i)}\ E(ze^{-j\omega_o}) + e^{-j(\Phi - \omega_o i)}\ E(z\ e^{j\omega_o})\right]$$

$$\therefore\ W_i(z) = -\frac{\alpha M}{2}\ U(z)\left[e^{j(\Phi - \omega_o i)}\ E(ze^{-j\omega_o}) + e^{-j(\Phi - \omega_o i)}\ E(ze^{j\omega_o})\right] \quad (2.48)$$

where $U(z) = 1/(z - 1)$

The output of the filter, y(n), is formed from

$$y(n) = \sum_{i=0}^{I-1} w_i(n)\ x(n-i) \quad (2.49)$$

where $x(n-i) = \cos(\omega_o n - \omega_o i)$

$$= \frac{1}{2}\left(e^{j\omega_o n}\ e^{-j\omega_o i} + e^{-j\omega_o n}\ e^{j\omega_o i}\right) \quad (2.50)$$

If we take the Z transform of each term in the summation for y(n), we have

$$Y(z) = \frac{1}{2} \sum_{i=0}^{I-1} \left[W_1(z\, e^{-j\omega_o})\, e^{-j\omega_o i} + W_1(ze^{j\omega_o})\, e^{j\omega_o i} \right]$$

$$Y(z) = -\frac{\alpha M}{4} \sum_{i=0}^{I-1} \Big\{ U(z\, e^{-j\omega_o})\, e^{-j\omega_o i} \Big[e^{j(\Phi - \omega_o i)}$$

$$E(z\, e^{-j2\omega_o}) + e^{-j(\Phi-\omega_o i)}\, E(z) \Big] + U(ze^{j\omega_o})\, e^{j\omega_o i}$$

$$\Big[e^{j(\Phi-\omega_o i)}\, E(z) + e^{-j(\Phi - \omega_o i)}\, E(z\, e^{j2\omega_o}) \Big] . \qquad (2.52)$$

$$\therefore \quad Y(z) = \frac{\alpha M}{4} \sum_{i=0}^{I-1} \Big\{ \Big[E(z) \Big[U(z\, e^{-j\omega_o}) e^{-j\Phi} + U(z\, e^{j\omega_o})\, e^{j\Phi} \Big] \Big]$$

$$+ \Big[E(z\, e^{-j\omega_o})\, U(z\, e^{-j\omega_o})\, e^{j(\Phi-2\omega_o i)} + E(z\, e^{j2\omega_o})\, U(z\, e^{j\omega_o})$$

$$e^{-j(\Phi - 2\omega_o i)} \Big] \Big\} \qquad (2.53)$$

The equation in the second square brackets contains terms of the form $e^{\pm j2\omega_o i}$ multiplied by a number of other terms which do not depend on i and can thus be taken outside the summation. Evaluating the summation of these exponential terms

$$\sum_{i=0}^{I-1} e^{\pm j2\omega_o i} = \frac{1 - e^{\pm j2\omega_o I}}{1 - e^{\pm j2\omega_o}} = e^{\pm j2\omega_o (I-1)}\, \frac{\text{Sin}\ (\omega_o I)}{\text{Sin}\ (\omega_o)} \qquad (2.54)$$

If the reference signal is synchronously sampled and the number of filter coefficients is equal to an integer (k) multiplied by half the number of samples per cycle,

$$I = k\pi/\omega_o$$

Therefore $\omega_o = k\pi/I$

$$\therefore \quad \frac{\text{Sin } (\omega_o I)}{\text{Sin } \omega_o} = \frac{\text{Sin } (k\pi)}{\text{Sin } (k\pi/I)} = 0 \qquad (2.55)$$

consequently the second term in the square brackets in the summation above is identically zero. We are left with I identical terms in E(z) and substituting for U(z), we obtain

$$Y(z) = -\frac{I \alpha M}{4} E(z) \left[\frac{e^{-j\Phi}}{z e^{-j\omega_o} - 1} + \frac{e^{j\Phi}}{z e^{j\omega_o} - 1} \right]$$

$$\therefore \quad \frac{Y(z)}{E(z)} = -\frac{I \alpha M}{2} \left[\frac{z \cos (\omega_o - \Phi) - \cos \Phi}{1 - 2z \cos (\omega_o) + z^2} \right] = G(z) \qquad (2.56)$$

This represents the transfer function of a linear, time invariant system. The secondary signal may be expressed in the z domain as

$$S(z) = C(z)\, Y(z) = C(z)\, G(z)\, E(z)$$

but $$E(z) = D(z) + S(z)$$

so $$\frac{E(z)}{D(z)} = \frac{1}{1 - C(z)\, G(z)} = H(z) \text{ say.} \qquad (2.57)$$

where H(z) is the transfer function between the error output and desired input, and the entire active controller acts as a linear time invariant system between these two signals.

Substituting for G(z) above and letting

$$\beta = \frac{I \alpha M}{2} \qquad \text{we obtain}$$

$$H(z) = \frac{1 - 2z \cos (\omega_o) + z^2}{1 - 2z \cos (\omega_o) + z^2 + \beta\, C(z) (z \cos (\omega_o - \Phi) - \cos \Phi)} \qquad (2.58)$$

This result is important because for a given error path, C(z), it allows the behaviour of the system to be determined analytically. Since no approximations have been made in the derivation of H(z), the complete behaviour of the system must be described by this behaviour.

This equation allows the conditions on the accuracy of the estimated secondary path, C, to be deduced in the limit of slow adaption. If the adaption is assumed to be very slow, i.e. $\beta \to 0$, a reordering of the transfer functions W and C becomes increasingly valid. The block diagram of the system now looks like figure 2.10a. If we redefine the reference signal as q in this figure, which is also a sinusoid at ω_0, and let $\epsilon(z) = \hat{C}(z)/C(z)$ be the error in the estimate of the error path, the block diagram becomes that in figure 2.10b. The transfer function in the error path has completely disappeared from this diagram so we can set $C(z) = 1$ in the equation above, however the error in the estimate of the secondary path remains as ϵ which has a phase response of Φ at ω_0. The transfer function of this system thus becomes

$$H(z) = \frac{1 - 2z\cos(\omega_0) + z^2}{1 - 2z\cos(\omega_0) + z^2 + \beta(z\cos(\omega_0 - \Phi) - \beta\cos\Phi)}$$

$$= \frac{1 - 2z\cos(\omega_0) + z^2}{z^2 - (2\cos(\omega_0) - \beta\cos(\omega_0 - \Phi))z + (1 - \beta\cos\Phi)} \qquad (2.59)$$

This is a second order recursive system whose stability can be investigated by examining whether the pole positions are within the unit circle. For small β, H(z) will have conjugate poles at a distance of $\sqrt{1 - \beta\cos\Phi}$ from the origin. Since all the terms in β are assumed positive, the distance of the pole from the unit circle can only be greater than 1 if cos Φ is negative, so the stability condition must be;

$$\cos\Phi > 0 \qquad \therefore \quad 90° > \Phi > -90° \qquad (2.60)$$

An additional condition for stability [23, p 169] is that

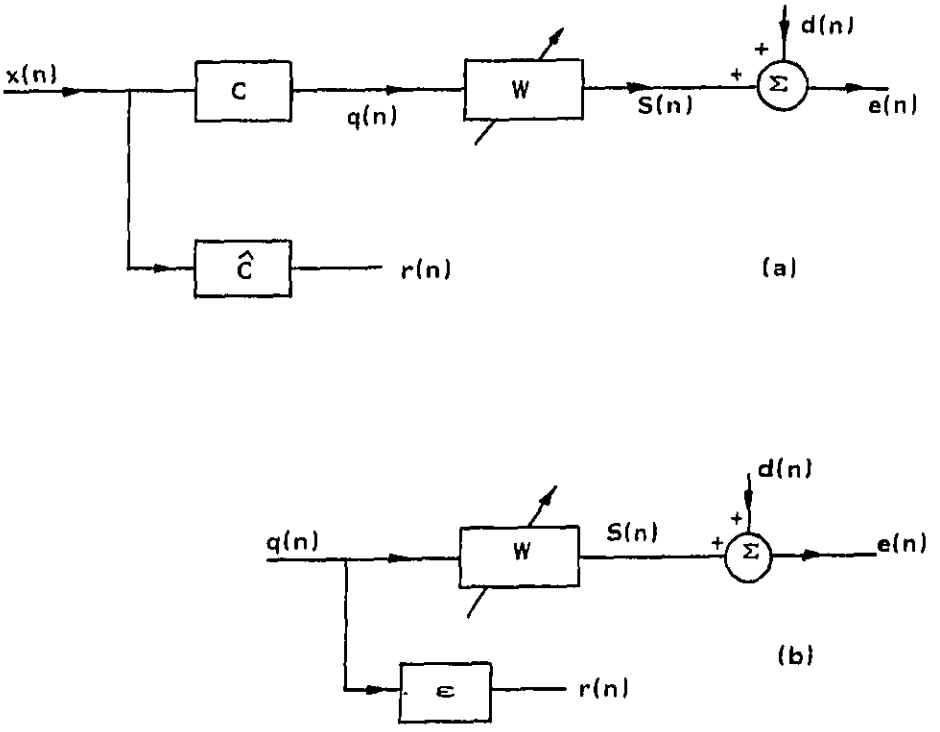

Figure 2.10. Block diagram for the adaptive system in the case of very slow adaption in (a) physical form and (b) reduced form.

$$| 2 \cos(\omega_0) - \beta \cos(\omega_0 - \Phi) | < 2 \qquad (2.61)$$

Since $\cos(\omega_0) > 0$ and β is assumed small, this condition must also be satisfied.

The time constant of convergence of an adaptive canceller with a sinusoidal reference but no extra transfer function in the error path is inversely proportional to α [20]. Assuming the adaption of the filtered x algorithm is already slow, to account for the dynamic properties of C, its convergence is further slowed if $\hat{C}$ is not a good match to C at ω_0. The analysis above indicates that the time constant of convergence is slowed down by a factor of $1/\cos\Phi$, where Φ is the phase difference between $_C$ and C at ω_0. If we consider only frequencies about ω_0 in the full block diagram of figure 2.10a, the error signal has been passed through a filter (C), with a magnitude response of approximately M at ω_0. The filtered reference

signal r(n) is also proportional to M, by definition. The magnitude of the update term, $\alpha\, e(n)\, r(n-i)$ will thus be a factor of M^2 larger than if C had not been present. This poses problems if multiple sinusoids are present in the reference signal since, in general, the modulus of the response of the filter C at each of these frequencies will be very different. Since only a single value of the convergence coefficient can be used, which applies to all the frequency components in the reference, this must be chosen so the system is stable for the frequency at which the response of C is largest. This will considerably slow down the convergence of the algorithm at frequencies where the response of C is small. Such behaviour is analogous to that due to the eigenvalue spread of the reference input discussed in Section 2.1.

2.6 Equivalent transfer function of the simulations

The simulations reported in section 2.4 were of a synchronously sampled system, which can be represented, using the results of sections 2.5, as equivalent transfer functions. In this section we compare the properties of these transfer functions with the behaviour observed in the simulations.

In the first simulation (corresponding to figures 2.5 and 2.6) above, we had:

$$\omega_o = \pi/2, \quad C(z) = z^{-10}, \quad \hat{C}(z) = z^{-2}$$

Thus $\cos(\omega_o) = 0$, $\Phi = \pi$, $\cos(\omega_o - \Phi) = 0$, $\beta = \alpha$, and the general transfer function of equation 2.58 reduces to

$$H(z) = \frac{1 + z^{-2}}{1 + z^{-2} + \alpha\, z^{-12}} \qquad (2.62)$$

This remarkably simple transfer function has zeros at $z = \pm j$ and poles corresponding to the roots of the denominator. This twelfth order equation cannot be solved analytically, but has been solved numerically for various values of α, and the resulting poles are plotted in figure 2.11. For very small values of α, one pair of

poles are almost coincident with the zeros at $z = \pm j$, and the other ten move rapidly away from the origin. For these small values of α, the behaviour is dominated by the poles and zeros near $\pm j$ and the transient response is that of a decaying exponential, as observed in figure 2.6(a). As α is increased the two pole pairs on the imaginary axis meet when $\alpha \simeq 0.07$, which corresponds approximately to figure 2.6(b). These pole pairs then break away from the imaginary axis and move outwards with increasing α. The transient behaviour in this region is dominated by these 4 poles each conjugate pair of which give a decaying sinusoidal response. These two decaying sinusoids beat together giving the behaviour observed in figure 2.6(c). The beat frequency gradually rises as β increases, and distance between the pairs of poles nearest $\pm j$ increases. For $\alpha \simeq 0.3$ these poles migrate outside the unit circle and the system becomes unstable.

Similar transfer functions can be obtained for other secondary delays, for example if the delay is one cycle of the reference signal, i.e. 4 sample periods, the transfer function becomes

$$H(z) = \frac{1 + z^{-2}}{1 - z^{-2} - \alpha z^{-6}} \qquad (2.63)$$

The trajectories of the 6 poles for this transfer function are shown in figure 2.12. These show a similar behaviour to those above except that only 4 poles move out from the origin. Apart from reinforcing the general interpretation given above, it is possible to solve for the pole positions exactly in this case by using z^2 as the variable in the denominator and solving the resulting cubic equation. Such an analysis shows that the roots of z^2 are real for $\alpha < 4/27$ but imaginary for values of α greater than this value. This values of α (0.148) thus corresponds to the point at which the poles on the imaginary axis meet, which agrees with the trajectories of figure 2.12.

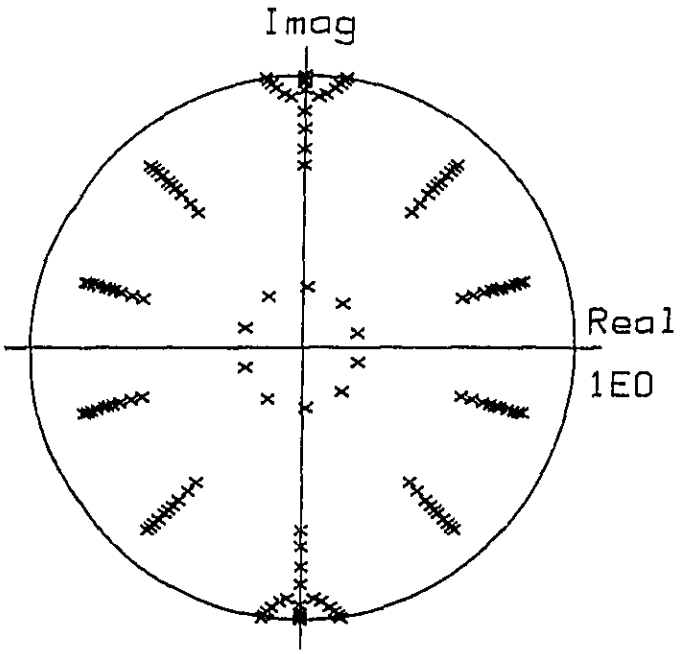

Figure 2.11 *Pole positions for the equivalent transfer function of the filtered x algorithm with a delay of 10 samples in the secondary path. With the convergence coefficient, α, very small there are ten poles evenly distributed about the origin, and two poles just inside the unit circle near z = ±j. The pole positions for α = 0.01, 0.02, 0.04, 0.06, 0.08, 0.10, 0.15, 0.20, 0.25 and 0.30 are also shown.*

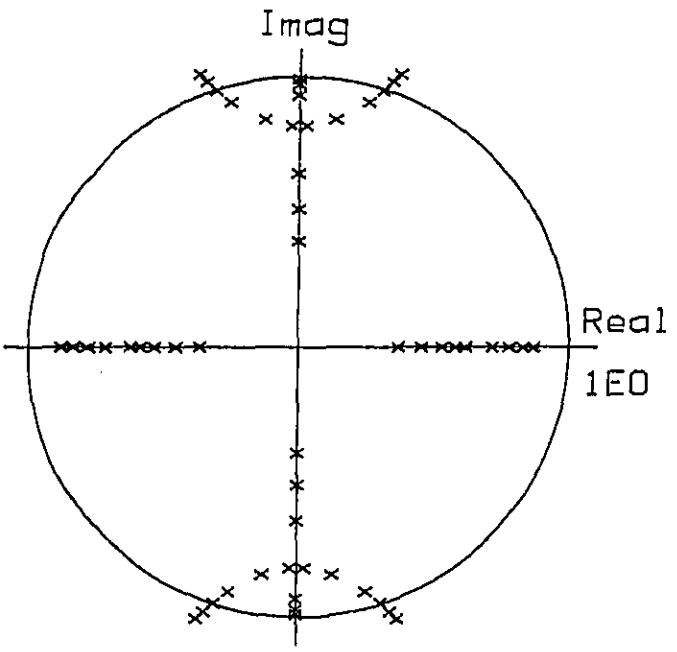

Figure 2.12 *Pole positions for the equivalent transfer function of the filtered x algorithm with a delay of 4 samples in the secondary path, for α = 0.02, 0.05, 0.10, 0.15, 0.2, 0.4, 0.6, 0.8 and 1.0.*

3. ADAPTIVE FILTERING IN MULTICHANNEL SYSTEMS

3.1. FORMULATION OF THE PROBLEM

We will assume that the physical system we are trying to control, and the transducers we are using, are linear. This assumption can be relaxed somewhat in practice, but is important as a starting point in the discussion. The primary field (acoustic or vibrational) present in the system without any active control is assumed to be correlated with an observable reference signal and also is assumed stationary, although again this assumption will be relaxed later on. This primary field is actively controlled using M actuators (typically loudspeakers or shakers) whose input is controlled so as to minimise the time averaged sum of the squares of L error sensors (typically microphones or accelerometers). This objective for the active control system is one which has been arrived at as a practical solution to the criterion of the minimisation of acoustic potential energy in an enclosure [24]. The placing of the actuators and error sensors is not discussed in this report, although it is noted that under certain conditions surprisingly few transducers need be used to achieve substantial reductions in total energy [24].

An equivalent block diagram of the active control system may be drawn if it is assumed that the contributions from the primary field to the outputs from the error sensors are derived from the reference signal. This assumption does not restrict the class of system being modelled, although the vector of transfer functions relating the reference signal to the error sensors ($\underline{A}$) has no obvious physical interpretation. The complete block diagram is shown in figure 3.1, where $\underline{W}$ represents the vector of M adaptive filters used to drive the secondary sources.

The signals described in the figure are entirely electrical. The electrical outputs from the error sensors and input to the secondary sources are the only signals used by the control system. Noting the assumption of linearity in the transducers as well as the physical system to be controlled, allows the use of the principle of superposition to derive an expression for the outputs of the error sensors. No separate consideration need be made of the interaction effects between the sources and sensors and the physical system. Loading for example does not need to enter the discussion.

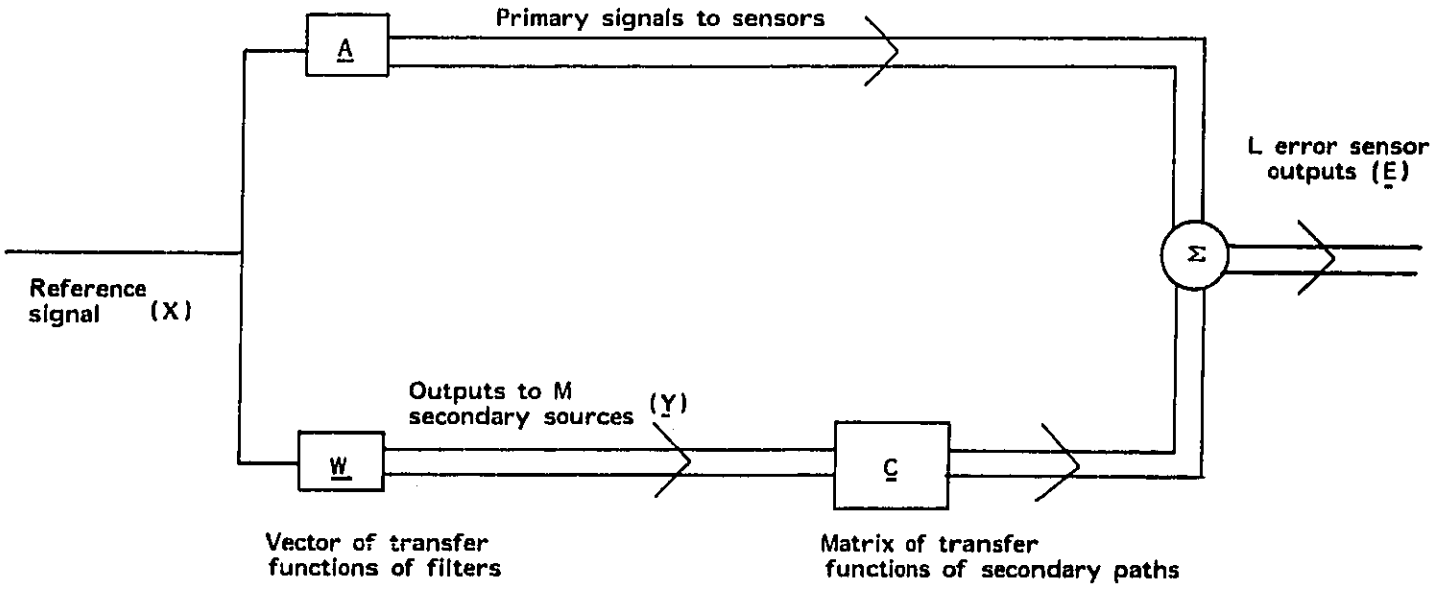

Figure 3.1 *Block diagram of a multivariable active control system.*

3.2. A MULTICHANNEL STOCHASTIC GRADIENT ALGORITHM

Let the sampled output of the ℓth error sensor be $e_\ell(n)$, which is equal to the sum of the primary desired signal from this sensor, $d_\ell(n)$, and the output due to each of the actuators. Let the sampled input to the mth actuator be $s_m(n)$, and the transfer function between this input and the output of the ℓth sensor be modelled as a Jth order FIR filter, whose jth coefficient is $c_{\ell mj}$, so that;

$$e_\ell(n) = d_\ell(n) + \sum_{m=1}^{M} \sum_{j=0}^{J-1} c_{\ell mj}\, s_m(n-j) \qquad (3.1)$$

It is assumed that there are L sensors and M actuators, and that $L \geq M$. Let the total error, E, be defined as;

$$E = \sum_{\ell=1}^{L} \overline{e_\ell^{\,2}(n)} \qquad (3.2)$$

where the bar indicates that a time average has been taken. If the reference signal, x(n), is at least partly correlated with each $d_\ell(n)$, it is possible to reduce E by driving the actuators with an FIR filtered version of the reference signal;

$$s_m(n) = \sum_{i=0}^{I-1} w_{mi}\, x(n-i) \qquad (3.3)$$

where w_{mi} is the ith coefficient of the filter driving the mth actuator.

The total error will be quadratic function of each of these filter coefficients, and the optimum set of filter coefficients required to minimise E may be evaluated adaptively using gradient descent methods. The gradient of the total error with respect to one coefficient is,

$$\frac{\partial E}{\partial w_{mi}} = 2 \sum_{\ell=1}^{L} \overline{e_\ell(n) \frac{\partial e_\ell(n)}{\partial w_{mi}}} \qquad (3.4)$$

Differentiating equation (3.1) using equation (3.3) gives

$$\frac{\partial e_\ell(n)}{\partial w_{mi}} = - \sum_{j=0}^{J-1} c_{\ell mj}\, x(n-i-j) \qquad (3.5)$$

This sequence is the same as the one which would be obtained at the ℓth sensor if the reference signal delayed by i samples were applied to the mth actuator. Let this be equal to $r_{\ell m}(n-i)$.

If each coefficient is now adjusted at every sample time by an amount proportional to the negative instantaneous value of the gradient, a modifed form of the well known LMS algorithm is produced;

$$w_{mi}(n+1) = w_{mi}(n) - \alpha \sum_{\ell=1}^{L} e_\ell(n)\, r_{\ell m}(n-i) \qquad (3.6)$$

Where α is the convergence coefficient.

For a single input, single output system (L=M=1), this corresponds exactly to the "filtered x LMS" algorithm discussed above.

A physical interpretation of the present algorithm can be obtained by considering the equivalent block diagram of the system; the block diagram of a simplified system is shown in figure 3.2a, with a single actuator producing an output y(n), which affects the outputs of two sensors, via the transfer function C_1 and C_2.

The two error signal $e_1(n)$ and $e_2(n)$ are the differences between these outputs and the desired signals $d_1(n)$ and $d_2(n)$. The output, y(n), is produced by driving the FIR filter W with the reference signal x(n). If the filter W is linear and time invariant, this block diagram is equivalent to the one in figure 3.2b, in which the signal $r_1(n)$ and $r_2(n)$ are produced by passing x(n) through C_1 and C_2 respectively.

If the n'th coefficient of the filter W were to be updated with the conventional LMS algorithm in either the upper or lower branches of figure 3.2b, the update terms would be of the form $e_1(n)\, r_1(n-1)$ or $e_2(n)\, r_2(n-2)$ respectively. If $d_1(n)$ and $d_2(n)$ were perfectly correlated with x(n)

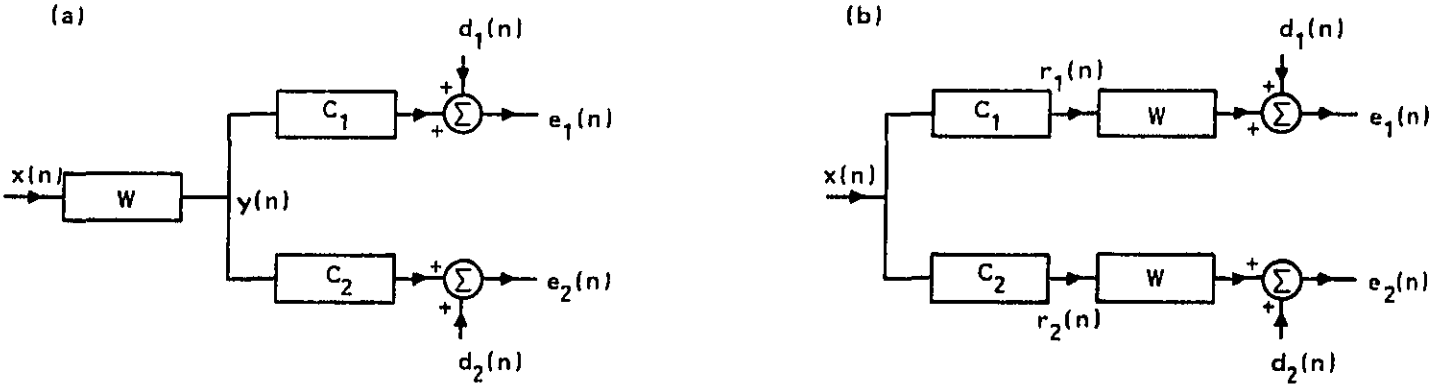

FIG. 3.2 BLOCK DIAGRAM OF AN ACTIVE CONTROL SYSTEM WITH ONE SECONDARY SOURCE AND TWO ERROR SENSORS, a) DIRECT, AND b) REARRANGED ASSUMING SLOW CONVERGENCE.

either $e_1(n)$ or $e_2(n)$ could be driven to zero by the action of the LMS algorithm operating in isolation on either the upper or lower branches. However what is required here is the minimisation of the sum of the mean square values of both errors, which in the steady state requires that the gradient of the total error with respect to each coefficient is zero. Updating the coefficients with the sum of the two update terms above achives this objective since the average value of the sum of the individual instantaneous gradients is equal to the sum of the average of these gradients.

3.3. COMPUTER SIMULATIONS OF THE ALGORITHM

A computer program has been written which simulates an active control system with four error sensors and two secondary actuators. A block diagram of the simulated system is given in figure 3.3. The program assumes a single frequency excitation which is synchronously sampled at exactly four samples per cycle, as above. If the primary field is assumed to be stationary its contribution to each of the four error sensors may also be derived via four two-point filters without loss of generality. In order to begin to model the dynamic response of the system realistically, both an overall delay, and some form of 'reverberant' behaviour must be included in the model of the response of the ℓ'th error sensor to the output of the m'th secondary source. This has been incorporated into the model by simulating the difference equation between the m'th source and the ℓth sensor as

$$e_\ell(n) = a_{\ell m}\, s_m(n - p_{\ell m}) + b_{\ell m}\, e_\ell(n - q_{\ell m}) \qquad (3.7)$$

The transfer function of which is

$$\frac{E_\ell(z)}{S_m(z)} = C_{\ell m}(z) = a_{\ell m}\, z^{-p_{\ell m}} \,/\, 1 - b_{\ell m}\, z^{-q_{\ell m}} \qquad (3.8)$$

This model has an overall delay of $p_{\ell m}$ samples, and the recursive term gives a first approximation to reverberant behaviour by causing past values of the error signal to recur in the output after $q_{\ell m}$ samples. The values of $a_{\ell m}$, $p_{\ell m}$, $b_{\ell m}$ and $q_{\ell m}$ for each of the eight filters are indicated in figure 3.3. They were chosen at random to be in the ranges 1 ± 0.5, 14 ± 4, 0.55 ± 0.24 and 6.5 ± 1.5 respectively. The average delay between a secondary source and error sensor is thus 14 samples or 3½ periods of the reference excitation.

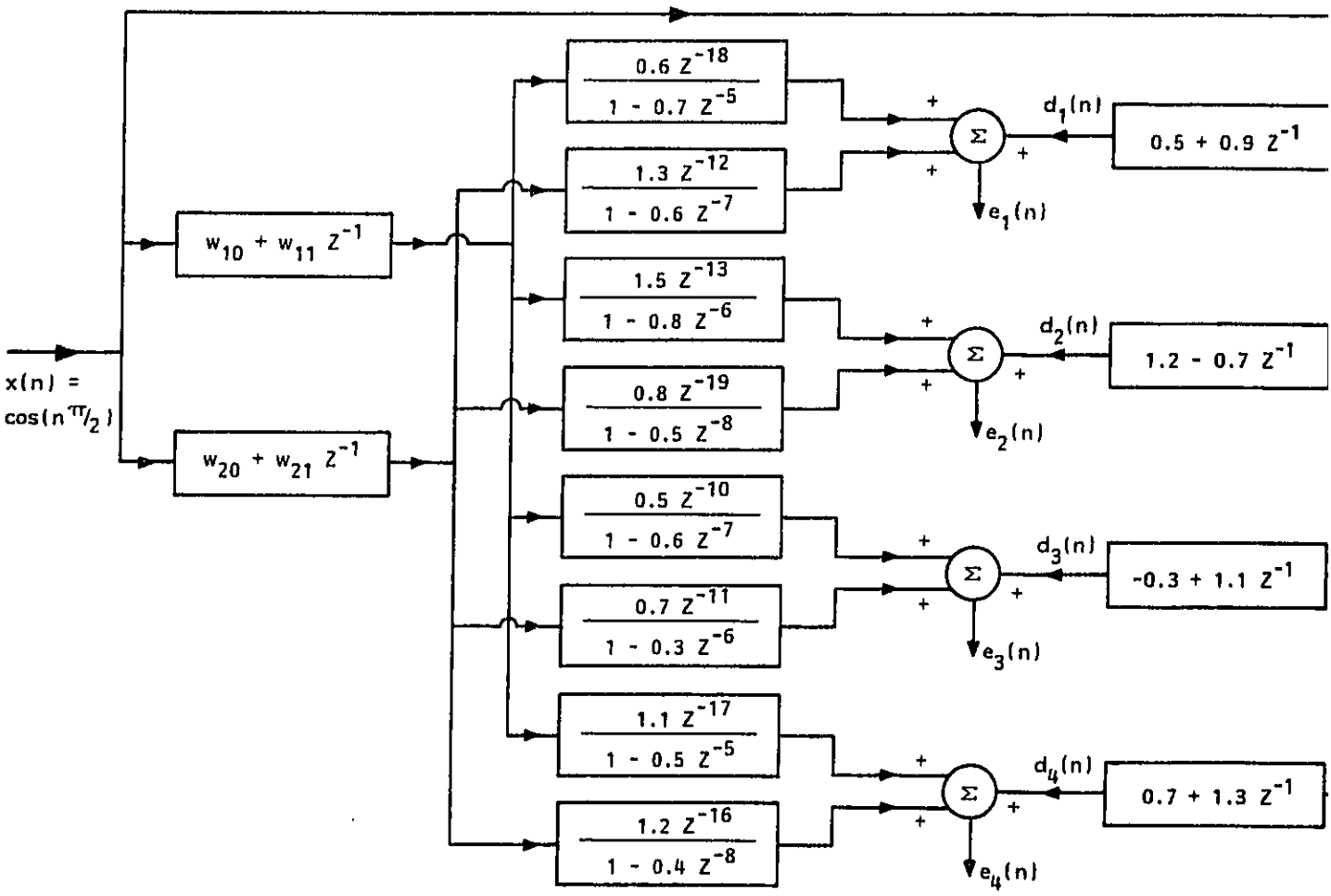

FIG. 3.3

BLOCK DIAGRAM OF THE SIMULATION PERFORMED WITH TWO SECONDARY SOURCES AND FOUR ERROR SENSO

The individual differences in the delays were introduced to simulate the difference in propagation time between variously positioned sources and sensors in a real system. The values chosen in the recursive terms of the difference equation means that the average time constant of the decay of their transient responses is about 10 samples or 2½ periods of the excitation. The desired signals were generated by passing x(n) through four two point FIR filters. The coefficients of these filters were of order 1, and the total error in the absence of any contribution from the actuators had a mean square value of 3.23. In order to generate the eight filtered reference sequences ($r_{gm}(n)$), x(n) was passed through eight two point FIR filters adjusted to have exactly the same magnitude and phase responses at the reference frequency as each of the filters defined by equation (3.8). Two adaptive FIR filters, with coefficients initially set to zero, were used to drive the two actuator outputs from the reference signal, and the coefficients of these filters were updated every sample using equation (3.6).

The way the total error (E) changed over 256 samples of the simulation, with a convergence coefficient (α) of 0.01, is presented in figure 3.4. The sum of the squares of all four error signals was smoothed using a two point moving average to obtain E. The value of this total error after 256 samples is within 3% of its values after several thousand samples. The trajectories of both coefficients of both adaptive filters over the same period are shown in figure 3.5.

A value of α of 0.01 gives approximately the fastest convergence time, even though some overshoot is present. If α is reduced to 0.005 this overshoot disappears and the algorithm converges monotonically, but slightly more slowly. The "overdamped" and underdamped" behaviour observed for the single channel algorithm with a pure delay (figure 2.6) is also observed in this multichannel case. This is illustrated in figure 3.6. The simulated system was unstable for values of α greater than about 0.05.

3.4. ADAPTION TIME OF THE ALGORITHM

One assumption implicitly made in the derivation of the algorithm was that the properties of the adaptive filters were varying slowly in comparison

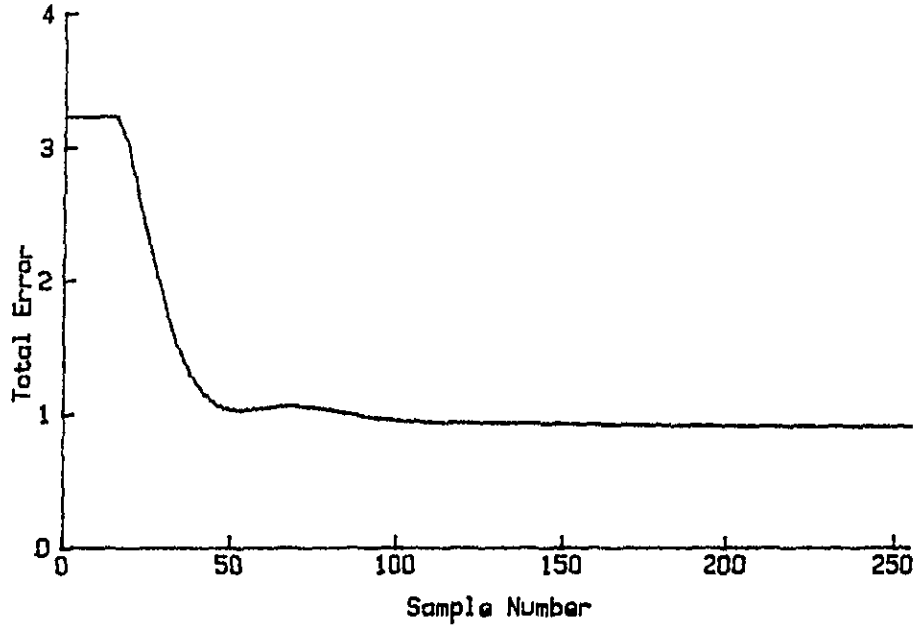

Figure 3.4 *The total, averaged, squared error from the four sensor for 256 samples of the simulation with* $\alpha = 0.01$.

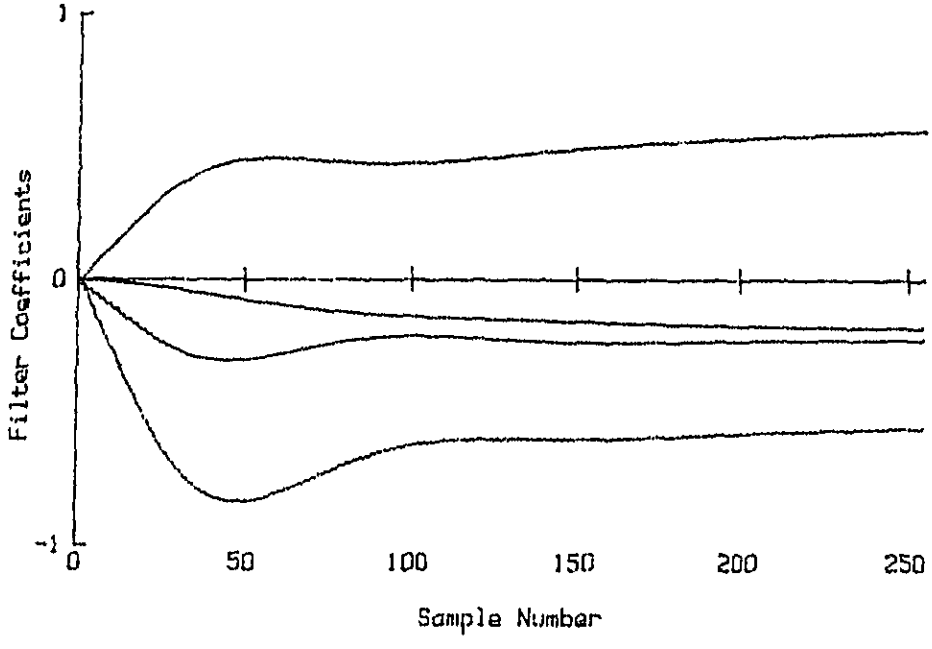

Figure 3.5 *The trajectories of the four adapting filter coefficients over 256 samples of the simulation with* $\alpha = 0.01$.

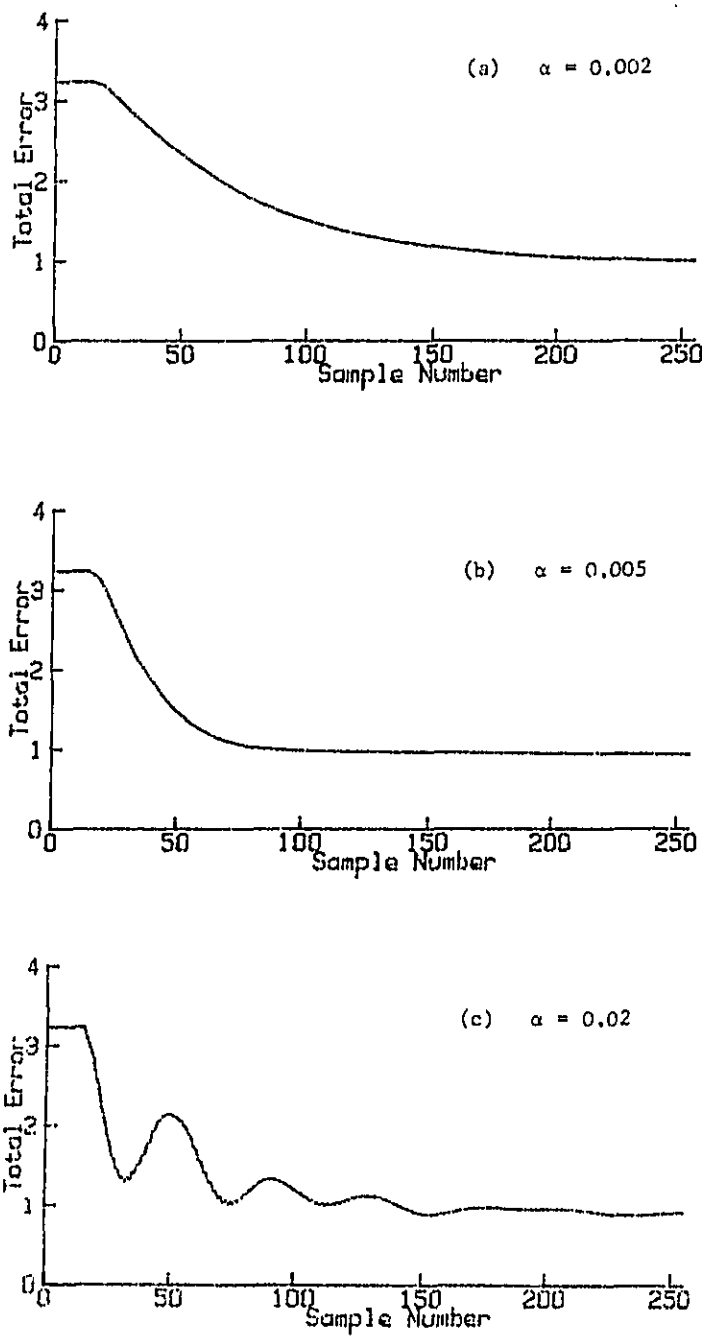

Figure 3.6 *Total, averaged, squared error at the four sensors against time for three values of the convergence coefficient: (a) α = 0.002, (b) α = 0.005, (c) α = 0.02*

with the timescale of the response of the system to be controlled. When the algorithm is implemented, however, the time constant of convergence, measured from the initial slope of figure 3.4. to be about 10 periods of the reference signal, is of the same order of magnitude as both the overall delays and the reverberant timescales used in the simulation. This reinforces the comment of Widrow and Stearns [6] that the filtered x LMS algorithm converges considerably faster than would apparently be expected, and extends this observation to the multichannel case. Another finding in the multichannel case is that there appears to be no more interference between the convergence of the coefficients of multiple filters than between the coefficients of a single filter, despite the fact that the update terms for each filter are coupled.

A further demonstration of the adaption time of the algorithm is afforded if the magnitude of the primary excitation at each of the error sensors is periodically modulated. This would, for example, be a simple represenation of two primary sources, e.g. propellers, operating at slightly different frequencies. It should be emphasised that the reference signals fed to the adaptive filters still has a constant amplitude.

The total error with and without the active control system in operation (using a convergence coefficient, α, of 0.01) is shown in figure 3.7. The period of the modulation in this simulation was 200 samples or 50 cycles at the excitation frequency. It can be seen that after a transient during the initial 100 samples or so, the adaptive filters are able to closely follow the changes in the primary excitation. This is confirmed by the trajectories of the coefficients over the same period which are plotted in figure 3.8.

3.5. ALGORITHM ROBUSTNESS

The algorithm has also been found to be very robust to errors in the generation of each of the filtered reference signals, $r_{\ell m}(n)$. In particular the algorithm can be made stable even with nearly 90° phase error in these signals, although the convergence parameter must be reduced somewhat to maintain stability in this case. This phase condition is intuitively reasonable in the case of slow convergence, since it implies that the average value of the individual terms in each update equation ($e_\ell(n)\ r_{\ell m}(n-i)$) must at least be of the correct sign for the error to be

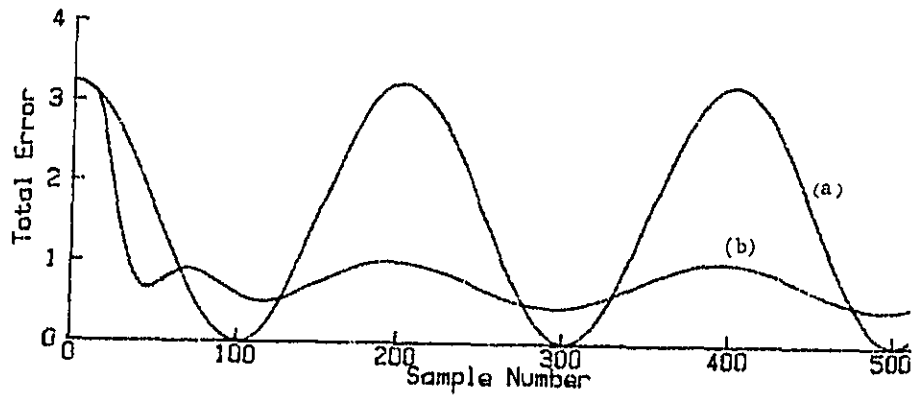

Figure 3.7 *Total, averaged, squared error from the four sensors when the primary source input is modulated, (a) with the adaption off, and (b) with the adaption on.*

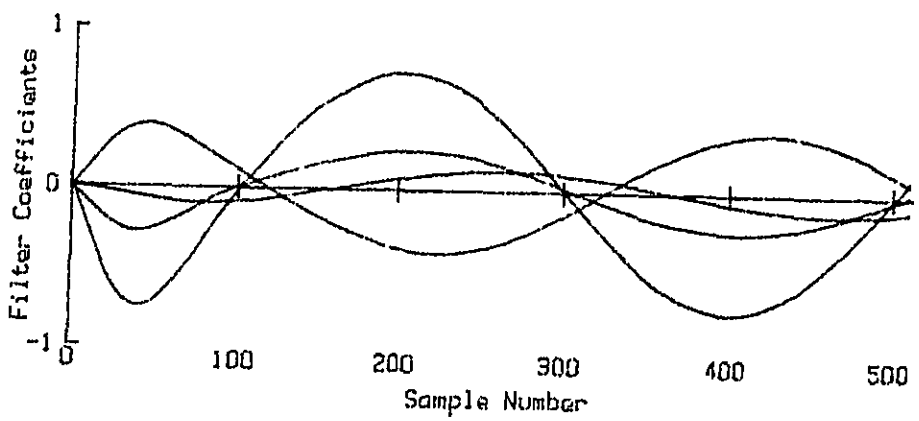

Figure 3.8 *The trajectories of the adapting filter coefficients for the case where the primary source input is modulated*

reduced during adaption and thus retain stability. The robustness of the algorithm is also demonstrated by other simulations which show that the convergence is largely unaffected by the introduction of either considerable uncorrelated observation noise or moderate non linearity in the transfer functions relating the sensor outputs to the actuator inputs.

3.6. THE STEADY STATE SOLUTION

Although the algorithm described above obviously achieves some reduction in the total mean square error, it is important to establish whether this is the optimal reduction which can be achieved with a given arrangement of sources and sensors. In the analysis presented below, the elements of all vectors and matrices are complex. The real and imaginary parts correspond to either the inphase and quadrature components of signals, or the real and imaginary parts of the controller transfer function, both evaluated at the single frequency of excitation. Since the sample rate has been chosen to be exactly four times the excitation frequency, the coefficients of the two point FIR filters used in the controller are the same as the real and the negative of the imaginary parts of the controllers transfer function.

Using nomenclature similar to that used in section 3.1, let

$$\underline{E} = [E_1\ E_2\ E_3\ \ldots\ E_L\ ]^T \qquad (3.9)$$

be the vector of complex error signals at ω_0.

$$\underline{Y} = [Y_1\ Y_2\ Y_3\ \ldots\ Y_M\ ]^T \qquad (3.10)$$

be the vector of complex outputs from the filters,

$$\underline{A} = [A_1\ A_2\ A_3\ \ldots\ A_L]^T \qquad (3.11)$$

be the vector of complex transfer functions between the reference signal and the error sensors,

$$\underline{W} = [w_1\ w_2\ w_3\ \ \ w_m]^T \qquad (3.12)$$

be the vector of complex transfer functions of the filters in the controller, and

$$\underline{C} = \begin{bmatrix} C_{11} & C_{12} & \cdots & \\ C_{21} & C_{22} & \ddots & \\ & & & C_{LM} \end{bmatrix} \qquad (3.13)$$

be the matrix of complex transfer functions between each secondary source and each error microphone.

The matrix equivalent of equation 3.1 in the frequency domain thus becomes

$$\underline{E} = \underline{A}\ X + \underline{C}\ \underline{Y} \qquad (3.14)$$

where x is the complex scalar reference signal. But $\underline{Y} = \underline{W}\ X$, so

$$\underline{E} = [\underline{A} + \underline{C}\ \underline{W}\ ]\ X \qquad (3.15)$$

The sum of the squares of the errors may be written in Hermitian quadratic form as

$$\sum_{\ell=1}^{L} |E_\ell|^2 = \underline{E}^H\underline{E} = [\underline{W}^H\underline{C}^H\underline{C}\ \underline{W} + \underline{W}^H\underline{C}^H\underline{A} + \underline{A}^H\underline{C}\ \underline{W} + \underline{A}^H\underline{A}]\ |X|^2 \qquad (3.16)$$

This may be minimised by setting $\underline{W}$ to:

$$\underline{W}_{opt} = -[\underline{C}^H\underline{C}\ ]^{-1}\underline{C}^H\underline{A} \qquad (3.17)$$

The error vector corresponding to this controller is thus

$$\underline{E}_{opt} = [\ \underline{A} + \underline{C}\ \underline{W}_{opt}]\ X \qquad (3.18)$$

and the optimum sum of error squared is

$$\underline{E}_{opt}^H\ \underline{E}_{opt} = [\underline{A}^H\underline{A} - \underline{A}^H\underline{C}\ [\underline{C}^H\underline{C}]^{-1}\ \underline{C}^H\underline{A}\ ]\ |X|^2 \qquad (3.19)$$

Table 3.1 Showing the set of filter coefficients and total mean square error calculated using matrix methods and achieved after various numbers of cycles in the simulation

Condition	W_{i0}	W_{i1}	W_{i2}	W_{i3}	Total Mean Square Error
Initial	0	0	0	0	3.235
Optimum	-0.022	0.667	-0.153	-0.467	0.887
After 200 cycles of simulation with α = 0.005	-0.179	0.599	-0.212	-0.536	0.908
After 3000 cycles of simulation with α = 0.005	-0.161	0.630	-0.194	-0.524	0.913

The elements of $\underline{A}$ are readily identifiable for the simulation performed above, from the block diagram of figure 3.3. The elements of the matrix $\underline{C}$ must be obtained from the transfer functions, $C_{\ell m}(z)$, listed in this figure by letting $z = e^{jw_0}$ where $w_0 = \pi/2$, so $z = j$.

Having identified the elements of $\underline{A}$ and $\underline{C}$, the expressions for $\underline{w}_{opt}$ and $\underline{E}^H_{opt}\underline{E}_{opt}$ were evaluated. The resulting set of optimal filter coefficients and the minimum mean square error are shown in table 3.1. Also shown in this table are the sets of filter coefficients and final total error observed in the simulations above after various numbers of cycles.

The first observation which can be made from this table is that the total error in the simulations never gets down to its minimum possible value, although it is within 2 or 3% of this value and so may be considered as achieving it for most practical purposes. The second point to note is that although the algorithm appeared to have reached a steady state in the simulations after several hundred cycles (figure 3.4.), it is in fact very slowly changing after this and appears to reach a completely "stable" state only after several thousand cycles. If the simulation is allowed to run for a further thousand cycles no change in the coefficients is observed. The total error after 3,000 cycles is in fact slightly greater than after 200 cycles. There is no obvious reason for this and it is probably due to finite precision effects in the computer. Of more direct interest is to observe the change in the filter coefficients from sample to sample after 3,000 cycles. Because none of the error signals are ever driven to zero the individual terms in the update equation (equation 3.6) are finite and

proportional to the squares of sinusoidal quantities with frequency 1/4. Although the average of the sum of these terms are zero in the steady state, the fact that an instantaneous estimate of the gradient is being taken means that the update term, although small, never completely disappears. In fact, the algorithm reaches an equilibrium in which the coefficients oscillate, about two stable sets of values, with a period of half a cycle. This effect is similar to the misadjustment errors discussed by Widrow and Stearns [6], which gives rise to a minimum error in normal adaptive filters which is slightly above the optimum.

3.7. MODIFICATIONS OF THE ALGORITHM

3.7.1. USE OF A MORE GENERAL COST FUNCTION

An error function or 'cost' function which is widely used in the field of optimal control [7] involves both mean square error terms and terms proportional to the mean square effort. For example if $y_m(n)$ is the output of the m'th filter, one cost function which could be used is:

$$J_T = \sum_{\ell=1}^{L} p_\ell \overline{e_\ell^2} + \sum_{m=1}^{M} q_m \overline{y_m^2} \qquad (3.20)$$

where p_ℓ and q_m are the weightings on the individual errors ($\overline{e_\ell^2}$) and 'efforts' ($\overline{y_m^2}$) respectively. The gradient of this cost function with respect to the i'th coefficient of the m'th filter is

$$\frac{\partial J_T}{\partial w_{mi}} = 2 \left[\sum_{\ell=1}^{L} p_\ell \, e_\ell(n) \, r_{\ell m}(n-i) + q_m y_m(n) \, x(n-i) \right] \qquad (3.21)$$

and the stochastic gradient algorithm for the adaptive filter becomes

$$w_{mi}(n+1) = w_{mi}(n) - \alpha \left[q_m y_m(n) \, x(n-i) + \sum_{\ell=1}^{L} p_\ell e_\ell(n) \, r_{\ell m}(n-i) \right] \qquad (3.22)$$

Note that the use of this cost function only adds one, easily calculated, term to the update equation for each coefficient.

The computer simulation of the multi-channel active control system described in section 3.3 was altered to incorporate a simplified form of this cost function: The values of all the error weighting coefficient, p_ℓ, were set equal (at unity) and the weighting function for both of the outputs (q_1 and q_2) were also set equal, at some variable value. If $q_1 = q_2 = 0$ the algorithm behaved as a section 3.3, as expected. As q_1 and q_2 were increased, the transient time of the algorithm did not appear to change, but the steady solution, after 600 cycles of the simulation, began to alter. For example, with $q_1 = q_2 = 10$ in the simulation the final mean square error was 1.7 (compared to 0.93 when $q_1 = q_2 = 0$, and 3.2 before adaption) and the final mean square value of the filter outputs was 0.127 (compared to 0.667 when $q_1 = q_2 = 0$). Consequently the algorithm allows much smaller secondary strengths to be used while still achieving some reductions in the error output.

Cost functions such as these have already been discussed, for example, in the higher harmonic control of helicopter vibration [11]. Their use would appear to be beneficial whenever there is a possibility of very large source strengths being necessary to achieve very small reductions at the error sensors, leading either to nonlinear behaviour or to increases in the total field away from the error sensors.

It is interesting to consider the scalar case of the LMS algorithm with this modified cost function, which may be writen in vector form as

$$\underline{w}_{n+1} = \underline{w}_n - \alpha[e(n)\,\underline{x}_n + q\,y(n)\,\underline{x}_n] \qquad (3.23)$$

substituting $y(n) = \underline{x}_n^T\,\underline{w}_n$ gives

$$\underline{w}_{n+1} = \left[\underline{I} - \alpha q\,\underline{x}_n\,\underline{x}_n^T\right]\underline{w}_n - \alpha\,e(n)\,\underline{x}_n \qquad (3.24)$$

In the case where ω_o is $\pi/2$ and two coefficient filters are being used, the 2 x 2 matrix $\underline{x}_n\underline{x}_n^T$ will, on average, be equal to 1/2 multiplied by the identity matrix (equation 2.36). The average behaviour

of the above algorithm is thus described by

$$\underline{w}_{n+1} = \gamma \underline{w}_n - \alpha e(n) \underline{x}_n \qquad (3.25)$$

where $\gamma = 1 - \alpha q/2 < 1$.

This implies that in the absence of any update term the value of the filter coefficients would gradually decay away. This expression is exactly the same as that described by Widrow and Stearns [6, p 377] as the 'leaky LMS' algorithm.

3.7.2. USE OF A WEIGHTED LEAST SQUARES CRITERION

It is sometimes desirable not to minimise the sum of the mean square values of a number of error signals, but to minimise the value of the largest one, the 'minimax' criterion. In general this minimisation problem is very nonlinear and thus difficult to solve analytically. However it has been suggested by Burrows and Shahinkaya [25] that a modified form of a least mean square algorithm could be used as an approximation to this in which the weightings on the individual errors are varied depending on their mean square value. Burrows and Shahinkaya used an iterative matrix inversion formulation to solve their equations, and adjust their error weighting values after each iteration. They found that the algorithm converged after two or three iterations.

A similar approach can be taken in the S.G. algorithm described above if the effort weighting functions (q_m) are set equal to zero, and the error weighting functions are made equal to the averaged squared value of the relevent error signal;

$$p_\ell = \overline{\overline{e_\ell^2}} \qquad (3.26)$$

A simulation has been performed in which the mean square errors were calculated using two point moving average filters;

$$\overline{\overline{e_\ell^2}} = \left[e_\ell^2 (n-1) + e_\ell^2 (n) \right]/2 \qquad (3.27)$$

and used to modify each p_ℓ, every sample, according to the equation above. The final results of this simulation were achieved after about 120 cycles and it was found that the value of the largest error signal after adaption was 0.47, compared to a value of 0.68 with all values of q_ℓ equal to unity. The mean square value of all the errors, however, increased to 1.08, from a value of 0.908 with all values of q_ℓ equal to unity.

In this simulation the maximum mean square error has been reduced by about 30% at the expense of an increase in the total mean square error of about 17%. It should, however, be noted that after convergence the other error signals have a mean square value well below the value of 0.47 quoted above. Consequently this is not a true minimax algorithm, which would drive the mean square values of all the errors to the same (minimum) value.

If the expression for p_ℓ in equation 3.26 is substituted into the original error criterion of equation 3.20 with all $q_m = 0$, it can be seen that the error criterion is the L_4 norm of the error. This in contrast to the L_2 norm used in the normal S.G. algorithm above and the L_∞ norm which must be used in a true minimax criterion. In fact higher order norms can be minimised, by taking $p_\ell = (e_\ell)^{2k}$, for example which would eventually minimise L_{2k+2}. A practical problem associated with such algorithms is the very slow convergence rate. This is due to the large difference in magnitudes of the individual terms of the coefficient update equation, a similar problem to that discussed at the end of section 2.5

REFERENCES

1. S.J. Elliott & P.A. Nelson, 1984, Technical Report 127.
   Models for describing active noise control in ducts.

2. A.D. White & D.G. Cooper, 1984, Applied Acoustics 17, 99-109.
   An adaptive controller for multivariable active noise control.

3. G.C. Goodwin & K.S. Sin, 1984, *Adaptive filtering, prediction and control,* Englewood Cliffs, New Jersey: Prentice Hall.

4. M.L. Honig & D.G. Messerschmitt, 1984, *Adaptive filters: Structures, algorithms and applications,* Dordrect, Netherlands: Kluwer Academic Publishers Group.

5. C.F.N. Cowan & P.M. Grant, 1985.
   Adaptive filters, Englewood Cliffs, New Jersey: Prentice Hall.

6. B. Widrow & S.D. Stearns, 1985.
   Adaptive Signal Processing, Englewood Cliffs, New Jersey: Prentice Hall.

7. R.J. Richards, 1979.
   An introduction to dynamics and control. New York, Longman.

8. S.J. Elliott & P.A. Nelson, 1985, Proceedings Internoise Conference.
   The active minimisation of sound fields.

9. A.J. Bullmore, A.R.D. Curtis, P.A. Nelson & S.J. Elliott, 1984, ISVR Contract Report 84/8.
   The active control of propeller induced cabin noise.

10. W. Johnson, 1982, NASA Technical Paper, 1996,
    Self tuning regulators for multicycle control of helicopter vibration.

11. M.W. Davis, 1984, NASA Contractors Reports 3821.
    Refinement and evaluation of helicopter real time self adaptive active vibration controller algorithms.

12. B. Widrow & M.E. Hoff, Jr., 1960, IRE WESCON Conv. Rec. Pt.4 96-104.
    Adaptive switching circuits.

13. Ljung, 1977, IEEE Transactions on automatic control AC22 551,
    Analysis of recursive stochastic algorithms.

14. P.M. Clarkson, 1983, Ph.D. Thesis, University of Southampton.
    Adaptive approaches to signal enhancement and deconvolution.

15. B. Widrow, 1970, in Aspects of Network and System Theory, R.E. Kalman and N. DeClaris (Eds), New York: Holt, Rinehart and Winston, 563 – 587.
Adaptive Filters.

16. S.S. Narayan et al, 1983 IEEE Transactions on acoustics speech and signal processing ASSP 31, 609-615, *Transform domain LMS algorithm.*

17. M. Dentino et al, 1978, Proceedings of the IEEE 66 (12) 1658-1659.
Adaptive filtering in the frequency domain.

18. C.W.N. Gritton and D.W. Lin, 1984 IEEE ASSP Magazine April 30-38.
Echo cancellation algorithms.

19. J.R. Glover, 1977 IEEE Transactions on acoustics speech and signal processing ASSP 25, 484-491.
Adaptive noise cancelling applied to sinusoidal interferences.

20. S.J. Elliott & P. Darlington, 1985, IEEE Transactions on acoustics, speech and Signal Processing ASSP-33 715-717.
Adaptive cancellation of periodic, synchronously sampled, interference.

21. P. Darlington & S.J. Elliott, 1985, Proceedings of the Institute of Acoustics 7 87-94.
Adaptive control of periodic disturbances in resonant systems.

22. J.C. Burgess, 1981, Journal of the Acoustical Society of America 70, 715-726.
Active adaptive sound control in a duct: a computer simulation.

23. M. Bellanger, 1984, *Digital processing of signals*, Chichester, John Wiley and Sons.

24. A.J. Bullmore et al, 1985, Proceedings of the Institute of Acoustics 7 55-63.
Active control of harmonic enclosed sound fields of low modal density: a computer simulation.

25. C.R. Burrows & M.N. Shahinkaya, 1983, Proceedings of the Royal Society, A386 77-94.
Vibration control of multi-made, motor-bearing systems.