Multiple Reference Active Noise Control

by

Yifeng Tu

Thesis Submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in Partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

Chris R. Fuller, Chairman
Ricardo A. Burdisso
William R. Saunders

March, 1997
Blacksburg, Virginia

Multiple Reference Active Noise Control

by
Yifeng Tu

Committee Chairman: Chris R. Fuller
Mechanical Engineering

Abstract

The major application of active noise control (ANC) has been focused on using a single reference signal; the work on multiple reference ANC is very scarce. Here, the behavior of multiple reference ANC is analyzed in both the frequency and time domain, and the coherence functions are provided to evaluate the effectiveness of multiple reference ANC.

When there are multiple noise sources, multiple reference sensors are needed to generate complete reference signals. A simplified method combines those signals from multiple reference sensors into a single reference signal. Although this method could result in satisfactory noise control effects under special circumstances, the performance is generally compromised. A widely adopted method feeds each reference signal into a different control filter. This approach suffers from the problem of ill-conditioning when the reference signals are correlated. The problem of ill-conditioning results in slow convergence rate and high sensitivity to measurement error especially when the FXLMS algorithm is applied. To handle this particular problem, the decorrelated Filtered-X LMS (DFXLMS) algorithm is developed and studied in this thesis.

Both simulations and experiments have been conducted to verify the DFXLMS algorithm and other issues associated with multiple reference ANC. The results presented herein are consistent with the theoretical analysis, and favorably indicate that the DFXLMS algorithm is effective in improving the convergence speed.

To take the maximum advantage of the TMS320C30 DSP board used to implement the controller, several DSP programming issues are discussed, and assembly routines are given in the appendix. Furthermore, a graphical user interface (GUI) running under Microsoft$^{®}$ Windows environment is introduced. The main purpose of the GUI is to facilitate parameters modification, real time data monitoring and DSP process control.

Acknowledgments

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Over the last two decades, much research has been conducted in the area of active noise control (ANC). The advantage of ANC lies in its effectiveness for reduction of low frequency noise. This is an ideal complement to the conventional passive noise control approach, which generally works efficiently at higher frequencies.

The basic philosophy of ANC has long been established, it can be dated back to 1936, when Paul Leug patented his ANC system in ducts [1]. The principle of ANC is the superposition of the two acoustical waves from the primary and secondary sources. When the two waves are out of phase and of the same amplitude, the superposition results in complete cancellation of the two waves and therefore generates a silent zone. Although the principle of ANC is straightforward, the real time implementation requires a fast, precise, and affordable controller, which is almost impossible to achieve with analogue devices. The recent advent of digital signal processing (DSP) technology has greatly propelled the development of various ANC systems.

There are two main approaches for ANC: feedback and feedforward. Olson and May first developed a feedback ANC system [2], the block diagram of which is illustrated in Fig.1.1a. The system uses an error sensor to detect the primary noise, and the detected noise is fed back to the controller to drive the secondary source located close to the error sensor. When the controller is optimally designed, a quiet zone can be created in the vicinity of the error sensor. The feedback approach has limited band width and possible instability caused by phase shift of the open-loop transfer function. The conventional feedback approach is non-adaptive, therefore careful design of the controller is needed to ensure the desired performance. An alternative feedback approach using an adaptive strategy has been proposed [3]. This approach differs from that of the feedforward in that it synthesizes a reference signal based on the filter output and the error signal.

Adaptive filters are advantageous over non-adaptive filters in that they can automatically search out optimum solutions, and keep track of the solutions when the environment changes. As a result, no elaborate efforts are required to design the filter coefficients in contrast to non-adaptive filters. Thus, feedforward ANC systems are always based upon adaptive filters. The block diagram of an ANC system using feedforward approach is illustrated in Fig. 1.1b. The system obtains a reference signal, and feeds it forward to the adaptive filter to control the secondary source. The reference signal must be coherent to the noise source in order to obtain desired noise reduction, this requirement should be contrasted to the feedback approach, in which no explicit reference signal is needed. A division should be drawn between the single frequency control and broadband control as they exhibit distinct characteristics. For the single frequency noise control, the future signal at the error sensor can be perfectly predicted through the adaptive filter, therefore no causality problem is involved in the design of the ANC system.
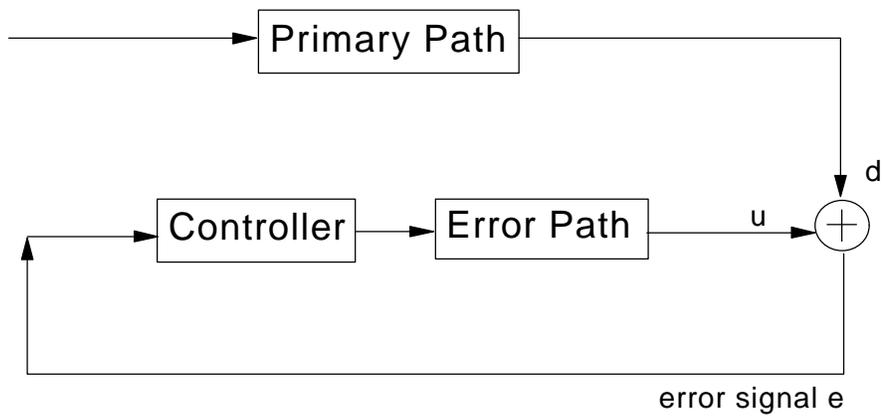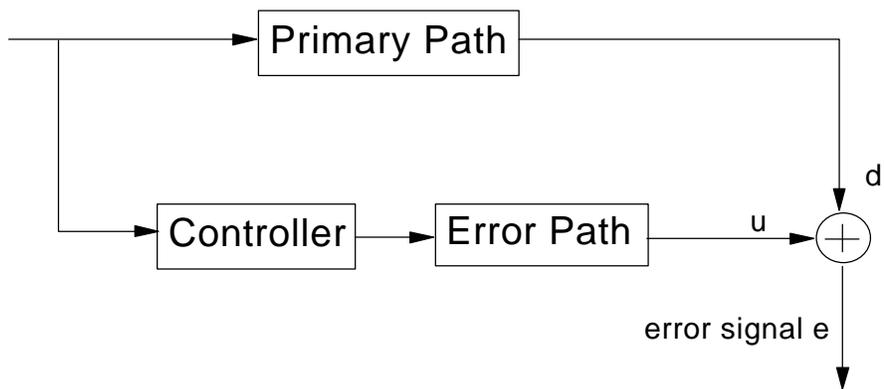
Figure 1.1 Two distinct control approaches for ANC, (a) feedback, (b) feedforward.

In contrast, the broadband noise control is restricted by causality. The reference signal must be obtained before the primary noise reaches the error sensor, this ensures enough time for the filter to generate the control signal and for the secondary acoustical wave to propagate to the error sensor. Although the adaptive filter is able to predict the future signal to a limited extent depending upon the disturbance spectrum, the penalty for non-causal system is dramatic deterioration of the performance.

The adaptive filter in a feedforward ANC system can be implemented with either IIR or FIR filters. The advantage of an IIR filter is due to its poles, which make it easier to match resonance, sharp cutoff, et al. However, an IIR filter is not always stable; when any one of its poles moves outside of the unit circle during the adaptive process, the IIR filter becomes unstable. In addition, small changes in the denominator coefficients may lead to large change in the frequency response if the poles are close to the unit circle, and the adaptation may converge to a local minimum since the performance surface of an IIR filter is generally non-quadratic. Thus, IIR filters have only limited applications. In some ANC systems, the feedback from the secondary source to the reference sensor is significant. To remove the feedback, the filtered-U algorithm with an IIR filter is proposed by Eriksson [4]. In the applications of the active structure acoustical control, to efficiently model the resonance characteristics of the structure, an IIR filter is also proposed to identify the error path [5]. However, ANC systems are mostly implemented with FIR filters, because, in contrast to IIR filters, FIR filters are always stable and their performance surfaces are always quadratic.

There are various adaptive algorithms for FIR filters. The selection among different algorithms is generally based on three factors [6]: computational complexity, performance and robustness. The computational complexity refers to the number of additions and multiplications, and the amount of storage required to implement the algorithm. A more complicated algorithm generally requires a faster DSP with larger memory area to carry out the real time computations, thereby increasing the cost of the ANC system. The performance is largely defined by: (1) convergence speed: the number of iterations required for the filter weights to converge, (2) misadjustment: the percentage deviation of the asymptotic mean square error to the optimum, (3) non-stationary tracking: the ability to track the optimum filter weights under a non-stationary environment. The three performance parameters are not independent, therefore it is impossible to satisfy critical requirements simultaneously. Which parameter should be given the highest priority depends on the specific application of an ANC system. Finally, the robustness refers to the performance variation due to measurement errors, finite precision calculations, et al. In general, an ideal algorithm delivering perfect results based on all three factors is not available, and a trade-off is inevitable when selecting an algorithm.

The ANC applications which have been successfully put into practice range from simple one dimensional acoustical field, e.g. air duct, to complicated three dimensional acoustical fields, e.g. aircraft cabin. The active control of complicated field requires that the secondary field be generated by a complex array of actuators so that the disturbance field can be matched [7]. In addition, minimizing the signal at a single error sensor may lead to increased noise level at other locations due to the variant spatial distribution of the noise field. In other words, the total

acoustical potential energy may increase even when the noise at the error sensor is reduced [8]. In order to achieve global noise control, it is necessary to use a number of secondary sources to minimize the mean square signals from a number of error sensors, which gives rise to a multiple input multiple output ANC system. A multiple error LMS algorithm is presented by Elliott et. al. for the active control of noise at a single frequency [9]. Later on, Elliott et. al. derived a steepest descent algorithm to minimize a cost function comprised of a combination of the sum of mean-square signals from a number of error sensors and the sum of the mean-square signals fed to the secondary sources [10]. Many others also contributed to the multiple channel ANC algorithms and associated issue [11][12].

It is noted that many noise fields are due to multiple uncorrelated or partially correlated disturbances. A typical example is the noise field inside a passenger vehicle where noise contributions are from tire-road interaction, engine firing, circulation fan, air stream and other factors. In addition, since the noise inside an vehicle has broadband nature, a feedforward approach is generally required. To gain maximum reduction of the noise field, it is necessary to obtain maximum coherence between the reference signal and the error signal. This implies the use of multiple reference sensors and associated signals. Therefore, in a complicated spatial noise field where there are several noise sources, a control strategy with multiple reference sensors, multiple secondary sources and multiple error sensors is needed to achieve desirable noise reduction [13]. Mikhael et. al discussed the noise coupling in a multiple source environment and studied optimal filter structures for multiple reference ANC [14]. Wallace et. al. proposed parallel adaptive filter structures for the multiple reference ANC inside a vehicle [15]. An average noise reduction of 16dB is reported. In principle, the number of reference sensors needed to achieve maximum coherence should be more than the number of noise sources. Traditional coherence techniques may be applied for noise source identification [16]. The effectiveness of coherence techniques depends not only upon the source coherence, but on the extent of measurement contamination [17]. Another criterion for the selection of reference sensors to detect noise sources has been proposed based on maximum potential control and the relative convergence rate [18]. A method to increase the convergence speed by using uncorrelators for the ANC in a multiple noise source environment has been presented by Masato et. al.[19]. In some other ANC systems, although there is only one major disturbance noise source, the systems exhibit strong non-linear phenomenon. In principle, a non-linear component can be treated as an independent noise source, and non-linearity is often a distributed phenomenon in ANC systems. Thus, these systems are equivalent to multiple noise source systems and are better treated using multiple reference sensors.

It is interesting to note that using multiple reference signals could potentially improve the causality of an ANC system as shown in Fig. 1.2. The ANC system shown in Fig. 1.2(a) is comprised of two noise sources. Using only the reference sensor 1 leads to the problem of a non-causal system, because the noise source 2 reaches the error sensor before it reaches the reference sensor 1. A similar problem arises using only the reference sensor 2. However, by using two reference sensors, shown in Fig. 1.2b, with one close to noise source 1, the other close to noise source 2, we can get complete coherent reference signals in advance, thus making the system causal. One might consider forming a single reference signal by summing signals from all the

reference sensors. However, this approach suffers from poor system performance, since essentially it is trying to model a multiple input multiple output system through a single input single output system.

Active Structure Acoustical Control(ASAC) [20] is a viable alternative to the conventional ANC. The ASAC principle is based on changing the radiation characteristics of a structure, i.e. the structure is driven by the controller to vibrate in inefficient acoustical radiation modes (non-volumetric mode) by applying active forces [21]. Since ASAC controls only efficient acoustical radiation modes (volumetric modes), the resultant advantage is reduced control efforts, and therefore fewer control actuators. This thesis illustrates the use of a multiple reference DFXLMS algorithm on an ASAC experiment.

This thesis is organized as follows: the next chapter (chapter 2) discusses a single channel feedforward active noise control system in both the frequency and time domain, and compares a few adaptive algorithms mainly in terms of computational complexity. Chapter 3 discusses the problems associated with multiple reference ANC; first with a single channel system, followed by a multiple channel system. Chapter 4 discusses the real time implementation of the DFXLMS algorithm; considerations are given to both DSP and host PC. Chapter 5 is dedicated to simulation and experimental results based on a vibrating plate. Finally, chapter 6 gives some conclusions and recommendations.

Noise source 1

Reference sensor 1

Error sensor

Controller

Secondary source

Reference sensor 2

Noise source 2

(a)

Noise source 1

Reference sensor 1

Secondary source

Controller

Controller

Error sensor

Reference sensor 2

Noise source 2

(b)
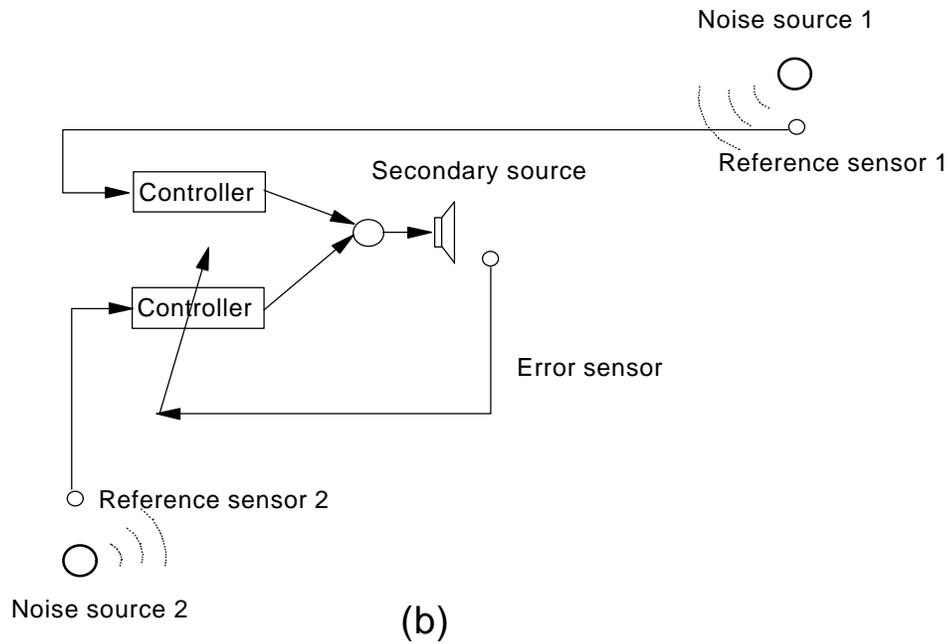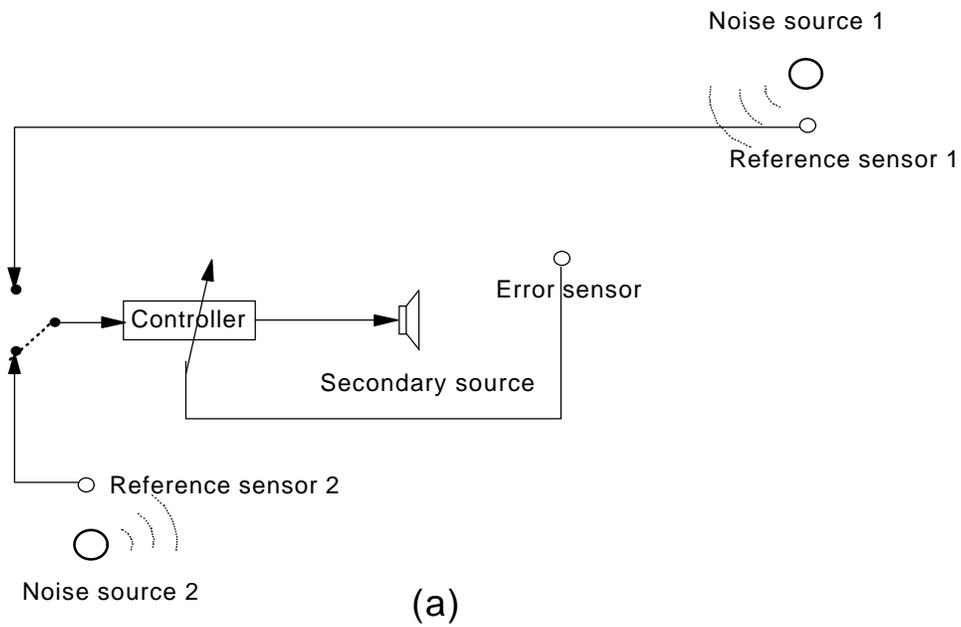
Figure 1.2 Multiple reference versus single reference, (a) single reference, (b) multiple reference.

2 Single Reference Single Channel ANC

This chapter discusses the principle of a single channel feedforward active noise control (ANC) system. First, the frequency response of an unconstrained optimum controller is derived, followed by the performance analysis of the corresponding ANC system in terms of relative noise reduction. The resultant frequency domain optimum controller is not necessarily realizable in physical system. In general, the controller in an ANC system is implemented with a causal FIR filter [22]. A FIR filter is said to be causal if the filter coefficients {h(n)} are zero when n is less than zero. Next, presented is a more practical approach in the time domain with a FIR filter as the basis of the controller, the corresponding optimum filter has a finite length and causal nature, thus, it is physically realizable. Finally, several adaptive algorithms are introduced and their performances are compared in terms of computational complexity and convergence rate.

2.1 Frequency Domain Analysis of Optimum Controller

Figure 2.1 illustrates a single channel ANC system. The signal from the error sensor e(w) is comprised of the contributions from both the primary noise source d(w) and the secondary control source y(w).

$$e(w) = d(w) + y(w) \tag{2.1.1}$$

where d(w) and y(w) are given by

$$d(w) = x(w)P(w) \tag{2.1.2}$$

$$y(w) = W(w)x(w)T(w) \tag{2.1.3}$$

where P(w), called the primary path, represents the transfer function from the noise source to the error sensor; and T(w), called the error path, represents the transfer function from the secondary source to the error sensor. Substituting equation (2.1.3) into equation (2.1.1) yields

$$e(w) = d(w) + W(w)x(w)T(w) \tag{2.1.4}$$

The optimum controller is designed such that the mean square error at every frequency is minimized. Thus, the cost function is constructed as

$$\mathsf{x}(w) = S_{ee}(w) = \mathrm{E}\big[e*(w)e(w)\big] \tag{2.1.5}$$

where the asterisk * denotes the complex conjugate. Substituting equation (2.1.4) into equation (2.1.5) yields

$$\mathsf{x}(w) = \mathrm{E}\big[(d*(w)d(w) + d*(w)W(w)x(w)T(w) + W*(w)x*(w)T*(w)d(w)$$
$$+ W*(w)x*(w)T*(w)W(w)x(w)T(w)\big] \tag{2.1.6}$$

The weight vector W is a complex variable, and the complex gradient vector is defined as

$$\nabla = \frac{\P\mathsf{x}(w)}{\P W(w)} = \frac{\P\mathsf{x}(w)}{\P W_R(w)} + j\frac{\P\mathsf{x}(w)}{\P W_I(w)} \tag{2.1.7}$$

Thus, the derivative of the cost function with respect to the transfer function W(w) can be expressed as
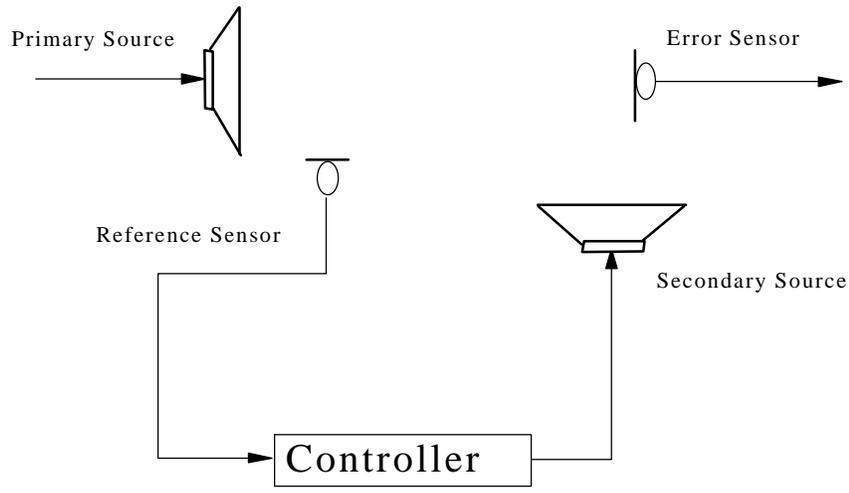
(a)
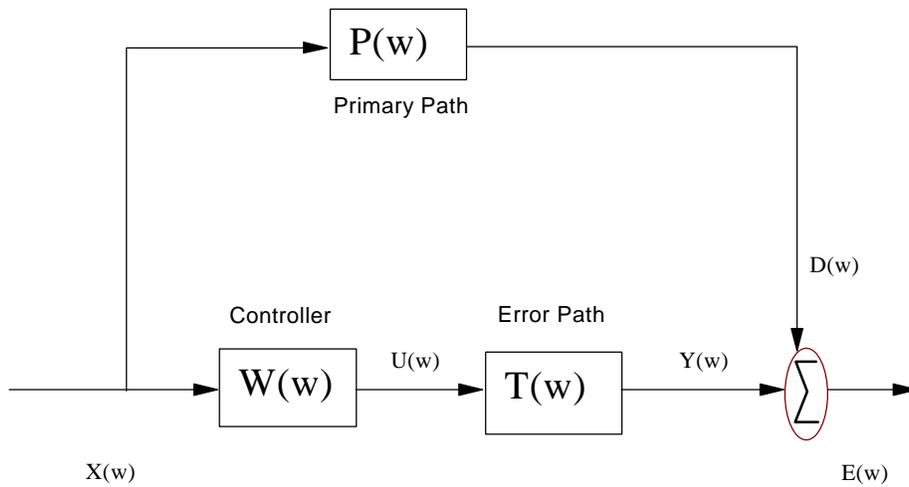
(b)

Figure 2.1 A single channel ANC system (a) simplified physical system; (b) frequency domain block diagram.

$$\frac{\P \chi}{\P W(w)} = 2\mathrm{E}\left[d(w)x*(w)T*(w) + W(w)x*(w)T*(w)x(w)T(w)\right] \qquad (2.1.8)$$

Setting equation (2.1.8) to zero, we obtain the optimum transfer function of the controller as

$$W_{opt}(w) = -\frac{\mathrm{E}\left[d(w)x*(w)T*(w)\right]}{\mathrm{E}\left[x*(w)T*(w)x(w)T(w)\right]} \qquad (2.1.9)$$

Substituting equation (2.1.2) into equation (2.1.9), we can further simplify the optimum transfer function as

$$W_{opt}(w) = -\frac{P(w)}{T(w)} \qquad (2.1.10)$$

Multiplying both sides of equation (2.1.10) by T(w) yields

$$W_{opt}(w)T(w) = -P(w) \qquad (2.1.11)$$

The above equation is intuitively appealing, as it indicates that the whole transfer function, i.e. the multiplication of optimum controller transfer $W_{opt}(w)$ and error path transfer function T(w), is exactly out of phase with and has the same amplitude as the primary transfer function P(w). In other words, the optimum controller transforms the reference signal into another signal which, at the error sensor, destructively interferes with the signal from the primary path.

On the other hand, the optimum transfer function can be obtained in terms of signal spectrums. The filtered reference signal is defined as

$$\hat{x}(w) = x(w)T(w) \qquad (2.1.12)$$

Substituting equation (2.1.12) into equation (2.1.9) results in

$$W_{opt}(w) = -\frac{\mathrm{E}\left[d(w)\hat{x}*(w)\right]}{\mathrm{E}\left[\hat{x}*(w)\hat{x}(w)\right]}$$

$$= -\frac{S_{d\hat{x}}(w)}{S_{\hat{x}\hat{x}}(w)} \qquad (2.1.13)$$

The above equation indicates the frequency response of the optimum controller is determined by the auto-spectrum of the filtered reference signal $S_{\hat{x}\hat{x}}(w)$ and the cross- spectrum between the filtered reference signal and the primary signal $S_{d\hat{x}}(w)$. It should be noted, although the above equation does not explicitly contain the system transfer functions, i.e. P(w) and T(w), the characteristics of the primary and secondary paths still determine the controller. In fact, the influence of the system characteristics is embodied in the auto-spectrum $S_{\hat{x}\hat{x}}(w)$ and the cross-spectrum $S_{d\hat{x}}(w)$.

When the controller is optimized using equation (2.1.13), the error signal reaches the minimum. The error signal, obtained in equation (2.1.6), can be rewritten as

$$S_{ee}(w) = S_{dd}(w) + S_{d\hat{x}}(w)W(w) + S_{\hat{x}d}(w)W*(w) + W*(w)W(w)S_{\hat{x}\hat{x}}(w) \qquad (2.1.14)$$

Substituting equation (2.1.13) into equation (2.1.14) results in

$$S_{ee}(w) = S_{dd}(w) + S_{d\hat{x}}(w)\frac{S_{\hat{x}d}(w)}{S_{\hat{x}\hat{x}}(w)} - S_{\hat{x}d}(w)\frac{S_{d\hat{x}}(w)}{S_{\hat{x}\hat{x}}(w)} - \frac{S_{d\hat{x}}(w)}{S_{\hat{x}\hat{x}}(w)}\frac{S_{\hat{x}d}(w)}{S_{\hat{x}\hat{x}}(w)}S_{\hat{x}\hat{x}}(w) \qquad (2.1.15)$$

Dividing equation (2.1.15) by $S_{dd}(w)$, the relative noise reduction is obtained as

$$\frac{S_{ee}(w)}{S_{dd}(w)} = 1 - \frac{S_{\hat{x}d}(w)S_{d\hat{x}}(w)}{S_{\hat{x}\hat{x}}(w)S_{dd}(w)}$$

$$= 1 - \gamma_{\hat{x}d}(w) \tag{2.1.16}$$

where $\gamma_{\hat{x}d}(w)$ is the *ordinary coherence function* between the filtered reference signal $\hat{x}(w)$ and the desired signal d(w). It can be shown that equation (2.1.16) still holds when there are measurement noise and unidentified noise sources. Equation (2.1.16) indicates that the relative noise reduction is upper bounded by $\gamma_{\hat{x}d}(w)$. It should be noted that $\gamma_{\hat{x}d}(w)$ is a function of frequency *w*, and an ANC system is typically focused on a certain frequency range. It is therefore not necessary to obtain a coherent reference signal throughout the entire frequency range. In fact, It is only essential to obtain a coherent reference signal at the particular frequency range where we are interested in achieving significant noise reduction.

The ordinary coherence function is unity if the signal d(w) is completely coherent with the reference signal x(w), which means that d(w) can be regarded as the output signal from a linear system with x(w) as the input signal. In this case, perfect noise cancellation can be achieved with the optimum controller defined by equation (2.1.13). However, the optimum controller defined in frequency domain may not be physically realizable, since the inverse Fourier transform of the frequency response function, i.e. the impulse response function, could be non-causal. For a causal filter, only the present and past input data are needed to generate the control output signal. This requirement is crucial for the broadband active noise control since no future reference signal is available at the time of processing to generate the control output signal.

2.2 Optimum FIR Filter

As discussed in the chapter 1, the controller in an active noise control system is typically implemented with a causal FIR filter. A typical structure of a causal FIR filter is shown in Figure 2.2, in which the filter length is M. The weight vector of the FIR filter can be represented by

$$\mathbf{w} = \begin{bmatrix} w_0 & w_1 & w_2 & \dots & w_{M-1} \end{bmatrix}^T \tag{2.2.1}$$

The input vector at time step k can be written as

$$\mathbf{x}(k) = \begin{bmatrix} x(k) & x(k-1) & x(k-2) & \dots & x(k-M+1) \end{bmatrix}^T \tag{2.2.2}$$

The filter output signal at time step k is the convolution of the input vector with the weight vector, i.e.,

$$u(k) = \mathbf{x}^T(k)\mathbf{w} = \mathbf{w}^T\mathbf{x}(k) \tag{2.2.3}$$

The error signal at time step k as shown in Figure 2.3 is composed of the primary source signal and the secondary source signal, which can be expressed as

$$e(k) = d(k) + y(k)$$

$$= d(k) + u(k)T(z) \tag{2.2.4}$$

where T(z) is the Z transform of the error path. The Z transform can be considered as a delayed operator, when a input signal x(k) passes through Z^{-1}, the delayed signal x(k-1) is obtained. Here,

no assumption is made about the structure of T(z), it could be in the form of either IIR or FIR filters. Substituting equation (2.2.3) into equation (2.2.4) yields

$$e(k) = d(k) + \mathbf{x}^T(k)\mathbf{w}T(z) \tag{2.2.5}$$

Defining the filtered reference signal as

$$\hat{\mathbf{x}}(k) = \mathbf{x}(k)T(z) \tag{2.2.6}$$

the derivative of the error signal with respect to the weight vector $\mathbf{w}$ is obtained

$$\frac{\P e(k)}{\P \mathbf{w}} = \hat{\mathbf{x}}(k) \tag{2.2.7}$$

Assuming that e(k), d(k), x(k) are statistically stationary and defining the cost function to be the mean square of the error signal, i.e.,

$$\chi = E\left[e^2(k)\right] \tag{2.2.8}$$

the gradient of the cost function with respect to weight vector is given by

$$\nabla = \frac{\P \chi}{\P \mathbf{w}} = 2E\left[e(k)\frac{\P e(k)}{\P \mathbf{w}}\right]$$

$$= 2E\left[d(k)\hat{\mathbf{x}}(k)\right] + E\left[\hat{\mathbf{x}}(k)\hat{\mathbf{x}}^T(k)\right]\mathbf{w}(k) \tag{2.2.9}$$

To obtain the minimum mean-square error, the gradient set to zero, i.e.

$$E\left[d(k)\hat{\mathbf{x}}(k)\right] + \mathbf{w}_{opt}E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right] = 0 \tag{2.2.10}$$

The above equation can be expressed more conveniently, if defining the square matrix $\mathbf{R}$ as

$$\mathbf{R} = E\left[\hat{\mathbf{x}}(k)\hat{\mathbf{x}}^T(k)\right]$$

$$= \begin{bmatrix} \hat{x}^2(k) & \hat{x}(k)\hat{x}(k-1) & ... & \hat{x}(k)\hat{x}(k-M+1) \\ \hat{x}(k-1)\hat{x}(k) & \hat{x}(k-1)\hat{x}(k-1) & ... & \hat{x}(k-1)\hat{x}(k-M+1) \\ \vdots & \vdots & \vdots & \vdots \\ \hat{x}(k-M+1)\hat{x}(k) & \hat{x}(k-M+1)\hat{x}(k-1) & ... & \hat{x}^2(k-M+1) \end{bmatrix} \tag{2.2.11}$$

and the vector $\mathbf{P}$ as

$$\mathbf{P} = E\left[d(k)\hat{\mathbf{x}}(k)\right]$$

$$= \begin{bmatrix} d(k)\hat{x}(k) & d(k)\hat{x}(k-1) & ... & d(k)\hat{x}(k-M+1) \end{bmatrix} \tag{2.2.12}$$

Equations (2.2.11) and (2.2.12) represent the auto-correlation matrix and the cross-correlation vector respectively. Substituting equation (2.2.11) and (2.2.12) into equation (2.2.10), the optimum weight vector is obtained as

$$\mathbf{w}_{opt} = -\mathbf{R}^{-1}\mathbf{P} \tag{2.2.13}$$

The above fixed optimum FIR filter (*Wiener filter*) is causal and of finite length, therefore it is realizable. However, to implement such a filter, the auto-correlation function of the filtered reference signal and the cross-correlation function between the filtered reference signal and the error signal have be obtained. This requirement implies the exact knowledge of both the plant and the signals, thus it substantially constrains the application. In many active noise control systems, the characteristics of the disturbance signals are statistically unknown, or the characteristics of the plant are time varying. To operate in such circumstances, the FIR filter should be able to tune itself automatically so that it tracks the optimum solution dynamically without the exact prior knowledge of the ANC system.
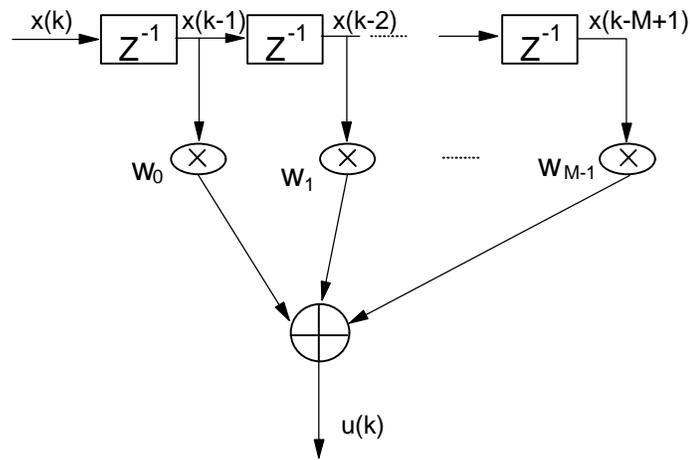
Figure 2.2 The structure of a FIR transversal filter.
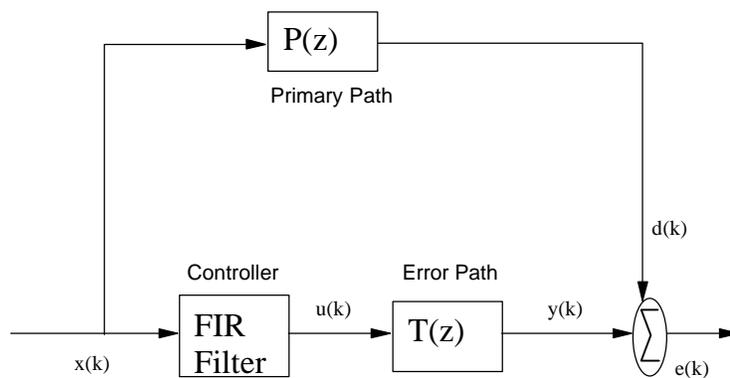
Figure 2.3 Block diagram of single channel ANC system with FIR Filter as the controller.

2.3 Adaptive Algorithm with a FIR Filter

In this section, the discussion is focused on a FIR filter with a transversal structure. The transversal structure , compared with other structures (for example, a lattice structure) has gained dominant attention in the application of active noise control due to its versatility and ease of implementation. In principle, adaptive signal processing is distinguished by two basic steps: (1) a signal filtering step, and (2) a filter weight updating step. The signal filtering step produces the output signal through the convolution of the input vector with the filter weight vector. The filter weight updating step marks an algorithm by its unique weight updating scheme. Three main adaptive algorithms are introduced beginning with the Filtered-X LMS algorithm, followed by the Filtered-X RLS algorithm. The Filtered-X LMS algorithm and the Filtered-X RLS algorithms are both the time domain algorithms. At the end, A brief discussion on the frequency domain block algorithm is also presented.

2.3.1 Filtered-X LMS Algorithm

The LMS algorithm [23] is an important member of the family of gradient based algorithms. The most significant advantage of the LMS algorithm lies in its simplicity and performance. When the LMS algorithm is applied to an ANC system , it is usually extended to the Filtered-X LMS algorithm due to the error path between the secondary source and the error sensor.

In section 2.2, the gradient of the mean square error signal with respect to the weight vector is obtained as

$$\nabla(k) = \frac{\partial \chi}{\partial \mathbf{w}} = 2\mathrm{E}\left[e(k)\frac{\partial e(k)}{\partial \mathbf{w}}\right]$$

$$= 2\mathrm{E}[e(k)\mathbf{x}(k)T(z)] \tag{2.3.1}$$

The above equation is based on the assumption that the error path transfer function T(z) is time invariant, or varies slowly compared with the weight vector updating. Such an assumption is valid for most applications. To calculate the expected value in equation (2.3.1), the discrete time average is used, which results in

$$\nabla(k) = 2\mathrm{E}[e(k)\mathbf{x}(k)T(z)] \cong \frac{2}{N}\sum_{i=0}^{N-1} e(k-i)\mathbf{x}(k-i)T(z) \tag{2.3.2}$$

The above equation requires too much computation. Further approximation of the expected value in equation (2.3.2) is obtained by using the instantaneous value to estimate the expected value, which is equivalent to setting N in equation (2.3.2) to unity. Thus, the gradient is rewritten as

$$\nabla(k) \cong 2e(k)\mathbf{x}(k)T(z) \tag{2.3.3}$$

The above approximation is the essence of the time domain LMS algorithm. It uses only the latest value of the reference signal and error signal to estimate the gradient, thus substantially simplifying the computation. the gradient method for updating the weight vector can be expressed as:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - m\overline{\nabla}(k) \tag{2.3.4}$$

Substituting equation (2.3.3) into equation (2.3.4) and ignoring the constant coefficient 2 yield

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu e(k)\mathbf{x}(k)T(z) \tag{2.3.5}$$

The filter output at time step k is the input vector convoluted with the weight vector, i.e.,

$$u(k) = \mathbf{x}^T(k)\mathbf{w}(k) \tag{2.3.6}$$

Equations (2.3.5) and (2.3.6) define the Filtered-X LMS algorithm. The equation (2.3.5) is different from the traditional LMS algorithm in that there is an extra term T(z), which represents the Z transform of the error path. Rewriting equation (2.2.6) yields

$$\hat{\mathbf{x}}(k) = \mathbf{x}(k)T(z) \tag{2.3.7}$$

Accordingly, equation (2.3.5) is adjusted as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu e(k)\hat{\mathbf{x}}(k) \tag{2.3.8}$$

In the above equation, the reference signal $\mathbf{x}$(k) is filtered through error path T(z) to form another signal $\hat{\mathbf{x}}$(k) for weight updating, hence the term *Filtered-X LMS* algorithm. It can be seen from equations (2.3.6) (2.3.7) and (2.3.8) that the Filtered-X LMS algorithm requires 2M+1+Z multiplications and M additions per iteration, where M is the number of filter weights, and Z is the number of multiplications required for obtaining the filtered reference. The Filtered-X LMS algorithm is summarized in Table 2.1.

Table 2.1 Summary of the Filtered-X LMS algorithm

Initialization:
$\mathbf{w}(0) = 0$
$e(0) = 0$
At each step
Weight Update:
$\hat{\mathbf{x}} = \mathbf{x}(k)T(z)$
$\mathbf{w}_M(k+1) = \mathbf{w}_M(k) - \mu e(k)\hat{\mathbf{x}}(k)$
Filter Output:
$y(k) = \mathbf{x}^T(k)\mathbf{w}(k)$

It should be noted that the weight vector converges to the optimum Wiener filter as defined in the previous section. However, the mean square error does not converge to the minimum mean square error; it is always greater than the minimum mean square error.

There are also many variant forms for the FXLMS algorithm, e.g. Normalized FXLMS algorithm, Leaky FXLMS algorithm, Signed FXLMS algorithm, variable step size FXLMS algorithm. These algorithms deliver better performance in terms of convergence rate, robustness, or computational complexity. In particular, the Leaky FXLMS algorithm, which minimizes the control effort as well as the error signal, is effective in avoiding the divergence problem due to the finite precision effects and the multiple solutions of an under-determined ANC system.

2.3.2 Filtered-X RLS Algorithm

The *recursive least square* (RLS) algorithm is a fundamental member of the family of least square based algorithms [24]. The Filtered-X RLS algorithm is a natural extension of the traditional RLS algorithm applied to an ANC system. An important feature of the RLS algorithm is that all the past input data, extending back to the initial time when the algorithm is started, are considered for the weight updating. This feature, compared to the Filtered-X LMS algorithm, provides better performance in terms of convergence rate and misadjustment. However, the improvement in performance is achieved at the expense of increased computational complexity.

The fundamental difference of the RLS algorithm compared to the LMS algorithm lies in the definition of the cost function, which is defined by

$$\xi(k) = \sum_{i=1}^{k} e^2(i)\lambda^{k-i} \tag{2.3.9}$$

where the parameter λ $(0<\lambda\leq1)$ is the weighting factor. λ is often set to 1 if the algorithm is operating in a stationary environment. A value of λ in the range $0.95<\lambda<0.9995$ has proved to be effective in tracking non-stationary signals [24]. The parameter e(i) is the error signal defined as

$$e(i) = d(i) + \mathbf{w}(k)^T \mathbf{x}(i)T(z)$$
$$= d(i) + \mathbf{w}(k)^T \hat{\mathbf{x}}(i) \tag{2.3.10}$$

where $\mathbf{w}$(k) is the weight vector at time step k with M coefficients, written as

$$\mathbf{w}(k) = \begin{bmatrix} w_0(k) & w_1(k) & w_2(k) & \ldots & w_{M-1}(k) \end{bmatrix}^T \tag{2.3.11}$$

and $\hat{\mathbf{x}}$(i) is filtered reference data vector at time step i, written as

$$\hat{\mathbf{x}}(i) = \begin{bmatrix} \hat{x}(i) & \hat{x}(i-1) & \hat{x}(i-2) & \ldots & \hat{x}(i-M+1) \end{bmatrix}^T \tag{2.3.12}$$

The gradient of the cost function with respect to weight vector is

$$\nabla = \frac{\P|\mathsf{X}(k)}{\P|\mathbf{w}(k)} = \sum_{i=1}^{k} |^{n-i} 2e(i)\hat{\mathbf{x}}(i) \tag{2.3.13}$$

Substituting equation (2.3.10) into equation (2.3.13) yields

$$\nabla = \sum_{i=1}^{k} |^{n-i} 2d(i)\hat{\mathbf{x}}(i) + \mathbf{w}(k)\sum_{i=1}^{k} |^{n-i} 2\hat{\mathbf{x}}(i)\hat{\mathbf{x}}(i) \tag{2.3.14}$$

The cross-correlation vector $\mathbf{P}$(k) is defined as

$$\mathbf{P}(k) = \sum_{i=1}^{k} |^{n-i} 2d(i)\hat{\mathbf{x}}(i) \tag{2.3.15}$$

and the auto-correlation matrix $\mathbf{R}$(k) is defined as

$$\mathbf{R}(k) = \sum_{i=1}^{k} |^{n-i} 2\hat{\mathbf{x}}(i)\hat{\mathbf{x}}(i) \tag{2.3.16}$$

Substituting equations (2.3.15) and (2.3.16) into equation (2.3.14) and setting the gradient to zero yield

$$\mathbf{w}(k) = \mathbf{R}^{-1}(k)\mathbf{P}(k) \tag{2.3.17}$$

15

It is inefficient and computational intensive to solve the matrix inversion in the above equation directly. Instead, the weight vector at time step k is obtained by adding the previous weight vector at time step k-1 with some adjustment vector. Thus equations (2.3.15) and (2.3.16) are obtained in recursive manner as

$$\mathbf{P}(k) = \mathsf{I}\,\mathbf{P}(k-1) + d(k)\hat{\mathbf{x}}(k) \tag{2.3.18}$$

$$\mathbf{R}(k) = \mathsf{I}\,\mathbf{R}(k-1) + \hat{\mathbf{x}}(k)\hat{\mathbf{x}}^T(k) \tag{2.3.19}$$

The *matrix inversion lemma* states [6] that

$$\text{if} \qquad \mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{C}^T \tag{2.3.20}$$

$$\text{then} \qquad \mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{I} + \mathbf{C}^T\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^T\mathbf{B} \tag{2.3.21}$$

The similarity between equations (2.3.19) and (2.3.20) allows us to apply the matrix inversion lemma to equation (2.3.19), i.e.

$$\mathbf{R}^{-1}(k) = \mathsf{I}^{-1}\mathbf{R}^{-1}(k-1) -$$

$$\mathsf{I}^{-1}\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(n)\big(1 + \hat{\mathbf{x}}^T(n)\mathsf{I}^{-1}\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(k)\big)\hat{\mathbf{x}}(k)\mathsf{I}^{-1}\mathbf{R}^{-1}(k-1) \tag{2.3.22}$$

Since the terms in the large brackets in equation (2.3.22) is scalar, it can be further rewritten as

$$\mathbf{R}^{-1}(k) = \mathsf{I}^{-1}\mathbf{R}^{-1}(k-1) - \frac{\mathsf{I}^{-2}\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(k)\hat{\mathbf{x}}^T(k)\mathbf{R}^{-1}(k-1)}{1 + \hat{\mathbf{x}}^T(k)\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(k)\mathsf{I}^{-1}} \tag{2.3.23}$$

The gain vector is defined as

$$\mathbf{G}(k) = \frac{\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(k)}{\mathsf{I} + \hat{\mathbf{x}}^T(k)\mathbf{R}^{-1}(k-1)\hat{\mathbf{x}}(k)} \tag{2.3.24}$$

Substituting equation (2.3.24) into equation (2.3.23) yields

$$\mathbf{R}^{-1}(k) = \mathsf{I}^{-1}\mathbf{R}^{-1}(k-1) - \mathsf{I}^{-1}\mathbf{G}(k)\hat{\mathbf{x}}^T(k)\mathbf{R}^{-1}(k-1) \tag{2.3.25}$$

The weight vector is now ready to be updated using equations (2.3.18) (2.3.19) and (2.3.25), i.e.

$$\mathbf{w}(k) = \mathbf{R}^{-1}(k)\mathbf{P}(k)$$

$$= \mathsf{I}\,\mathbf{R}^{-1}(k)\mathbf{P}(k-1) + d(k)\mathbf{R}^{-1}(k)\hat{\mathbf{x}}(k)$$

$$= \mathbf{R}^{-1}(k-1)\mathbf{P}(k-1) - \mathbf{G}(k)\hat{\mathbf{x}}^T(k)\mathbf{R}^{-1}(k-1)\mathbf{P}(k-1) + d(k)\mathbf{R}^{-1}(K)\hat{\mathbf{x}}(k)$$

$$= \mathbf{w}(k-1) - \mathbf{G}(k)\mathbf{x}^T(k)\mathbf{w}(k-1) + d(k)\mathbf{G}(k)$$

$$= \mathbf{w}(k-1) + \mathbf{G}(k)\big(d(k) - \hat{\mathbf{x}}^T(k)\mathbf{w}(k-1)\big)$$

$$= \mathbf{w}(k-1) + \mathbf{G}(k)e(k) \tag{2.3.26}$$

where e(k) is called the priori estimation error, and is given by

$$e(k) = \big(d(k) - \hat{\mathbf{x}}^T(k)\mathbf{w}(k-1)\big) \tag{2.3.27}$$

Equation (2.3.26) is similar to the weight update equation for the LMS algorithm shown in equation (2.3.8), in which $\mathsf{m}\hat{\mathbf{x}}(k)$ is replaced by the gain vector $\mathbf{G}(k)$. The Filtered-X RLS Algorithm is summarized in Table 2.2.

The Filtered-X RLS algorithm is much more complicated than the Filtered-X LMS algorithm since the computational complexity is on the order of N^2, where N is the number of the filter weights. However, the RLS algorithm is characterized by fast convergence rate, and it is relatively insensitive to the eigenvalue spread of the auto-correlation matrix of the input data.

Furthermore, some fast algorithms [25] based on the RLS algorithm are able to achieve a computational complexity which increases linearly with the number of filter weights. However, compared with the LMS algorithm, they still require much more computation.

Table 2.2 Summary of Filtered-X RLS algorithm

Initialization:
$\qquad \mathbf{D}(0) = d^{-1}I \qquad\qquad \delta$ small positive constant
$\qquad \mathbf{w}(0) = 0$
$\qquad e(0) = 0$
At each time step:
$\qquad$ Weight Update
$\qquad \hat{\mathbf{x}}(k) = \mathbf{x}(k)T(Z)$
$\qquad \mathbf{A}(k) = \mathbf{D}(k-1)\hat{\mathbf{x}}(k)$
$\qquad b(k) = l + \hat{\mathbf{x}}^T(k)\mathbf{D}(k-1)\hat{\mathbf{x}}(k)$
$\qquad \mathbf{G}(k) = \dfrac{\mathbf{A}(k)}{b(k)}$
$\qquad \mathbf{D}(k) = \dfrac{1}{l}\Big(\mathbf{D}(k-1) - \mathbf{G}(k)\mathbf{x}^T(k)\mathbf{D}(k-1)\Big)$
$\qquad \mathbf{w}(k) = \mathbf{w}(k-1) + \mathbf{G}(k)e(k)$
$\qquad$ Filter Output:
$\qquad y(k) = \mathbf{x}^T(k)\mathbf{w}(k)$

2.3.3 Block Algorithm and the Frequency Domain Implementation

For both the LMS and RLS algorithms, the weight vector is updated at every sample interval. Alternatively, the weight vector can be updated block by block, each block contains a number of data depending on the block length, thus the term *block algorithm* [26]. The filter output and the weight vector are not computed immediately upon receiving the input data. Instead, the computation starts when the input data are accumulated into a block. During each block, the weight vector is set to be constant. The block algorithm is usually implemented in the frequency domain. There are two primary advantages for the frequency domain block algorithm. First, when implemented with the fast Fourier transform (FFT), the computational complexity can be significantly reduced compared to the Filtered-X LMS algorithm, especially when the number of filter coefficients is large. Second, the signal after FFT is almost uncorrelated, which implies that the auto-correlation function $R(t)$ be zero if τ is not zero. As a result, a different or time-varying step size can be assigned for each weight, therefore allowing a uniform convergence rate for all the filter weights.

For each block k, the output vector, weight vector, and input matrix can be rewritten as

$$\mathbf{y}(k) = \begin{pmatrix} y(kL+0) \\ y(kL+1) \\ y(kL+2) \\ \cdots \\ \cdots \\ y(kL+L-1) \end{pmatrix}_{L \times 1} \tag{2.3.28}$$

$$\mathbf{w}(k) = \begin{pmatrix} w_0(k) \\ w_1(k) \\ w_2(k) \\ \cdots \\ \cdots \\ w_{N-1}(k) \end{pmatrix}_{N \times 1} \tag{2.3.29}$$

$$\mathbf{X}(k) = \begin{bmatrix} x(kL+0) & x(kL+1) & \cdots & x(kL+L-1) \\ x(kL-1) & x(kL+0) & \cdots & x(kL+L-2) \\ x(kL-2) & x(kL-1) & \cdots & x(kL+L-3) \\ \vdots & \vdots & \vdots & \vdots \\ x(kL-N+1) & x(kL-N+2) & \cdots & x(kL+L-N) \end{bmatrix}_{N \times L} \tag{2.3.30}$$

where L is the block length, and N is the filter length. In order to take the best advantage of the block algorithm, the block length L is usually set to the filter length N. The above three equations are related by

$$\mathbf{y}(k) = \mathbf{X}^T(k)\mathbf{w}(k) \tag{2.3.31}$$

and the error vector can be obtained as

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k)T(Z)$$
$$= \mathbf{d}(k) - \hat{\mathbf{X}}^T(k)\mathbf{w}(k) \tag{2.3.32}$$

The cost function is defined as the average mean square error, that is

$$\mathsf{x}(k) = \frac{1}{L}\mathrm{E}\big(\mathbf{e}^T(k)\mathbf{e}(k)\big) \tag{2.3.33}$$

The gradient of the cost function with respect to the weight vector is

$$\nabla = \frac{\P \mathsf{x}(k)}{\P \mathbf{w}(k)} \cong \frac{2}{L}\hat{\mathbf{X}}^T(k)\mathbf{e}(k) \tag{2.3.34}$$

The weight vector is defined as follows

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{L}{2}\mathsf{m}\nabla$$
$$= \mathbf{w}(k) - \mathsf{m}\mathbf{X}^T(k)\mathbf{e}(k) \tag{2.3.35}$$

18

Equations (2.3.31) and (2.3.35) basically define the block LMS (BLMS) algorithm. At this point, the computational complexity of the LMS and the BLMS algorithm is essentially the same, although the BLMS algorithm requires much more memory for storing data. The advantage of the BLMS algorithm lies in the frequency domain implementation [27][28]. As shown in Figure 2.4, the reference signal x(n) is transformed into the frequency domain signal X(n) using the FFT. The frequency domain (complex) filter accepting the transformed reference signal X(n) as input produces the frequency domain output signal Y(n), which is further transformed back to the time domain signal y(n) using inverse FFT to drive the secondary source. The error signal is also transformed into the frequency domain for the weight updating of the complex filter.

It is important to note that the multiplication in the frequency domain essentially corresponds to a circular convolution in the time domain, and an adaptive FIR filter produces only the linear convolution. Thus, some data constraints must be enforced in order to obtain the desired linear convolution. The overlap-save and overlap-add methods [29] are generally adopted for applying the data constraints.
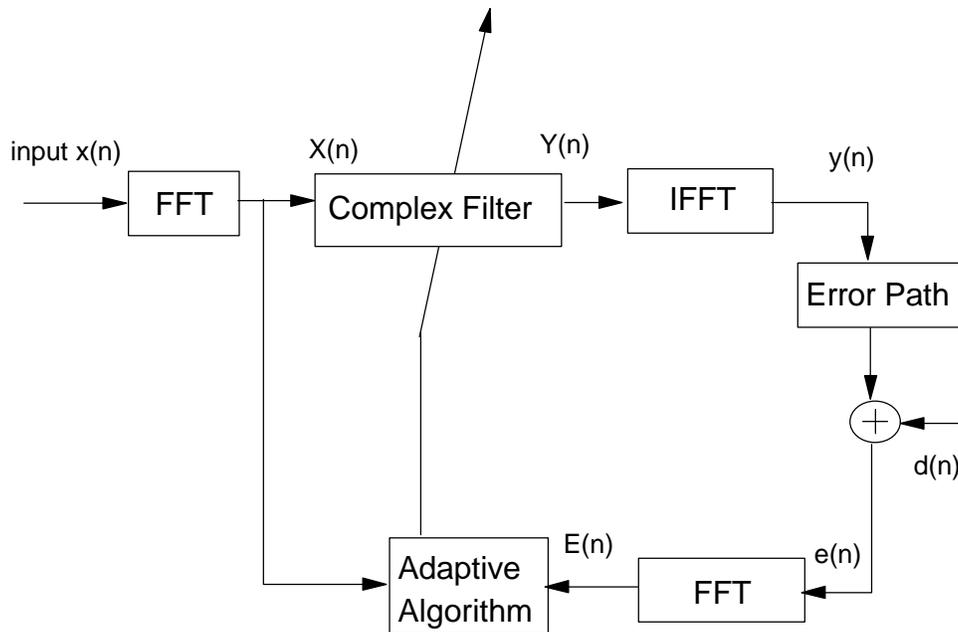
Figure 2.4 Frequency domain adaptive filter configurations.

3 Multiple Reference ANC

In a complicated noise field when there are multiple noise sources, e.g. car interior noise, multiple reference signals are required to obtain satisfactory noise reduction effect. This chapter discusses the behavior of a multiple reference ANC system. The first section is focused on a single channel (i.e. single secondary source, single error sensor) system. First, the frequency domain optimum solution is derived, and the coherence analysis is applied to show how the multiple reference approach could potentially improve the performance. Next, the correlation matrix of the reference signals is examined, and two examples are given to illustrate that the system is ill-conditioned if the reference signals are correlated. Moreover, under the extreme circumstance when any two reference signals are linearly dependent, the ANC system becomes underdetermined, i.e. the solution to the system is not unique. To deal with the problem of ill-conditioning, an adaptive algorithm to decorrelate the reference signals is presented.

The application of multiple reference ANC is mainly focused on complex systems with complicated noise sources and acoustical fields. Such a system generally requires multiple control channels to achieve spatial noise reduction effect. Therefore, the analysis is extended to a multiple channel (multiple secondary sources and multiple error sensors) control system.

3.1 Single Channel Control System

A single channel MRANC system is considered, as shown in Figure 3.1, in which there are K reference signals, one control signal, and one error signal.

3.1.1 Frequency Domain Optimum Solution

The primary signal d is due to multiple noise sources passing through multiple primary paths. The signal e at the error sensor is contributed from both the primary signal and the control signal, and can be written as

$$e(w) = d(w) + T(w)u(w) \qquad (3.1.1)$$

where T(w) is the transfer function between the secondary source and the error sensor, d(w) is the primary signal due to the multiple noise sources, and u(w) is the summation of all the filter output signals, given by

$$u(w) = \sum_{i=1}^{K} x_i(w)W_i(w) \qquad (3.1.2)$$

where K is the number of reference signals. Substituting the above equation into (3.1.1) results in

$$e(w) = d(w) + T(w)\sum_{i=1}^{K} x_i(w)W_i(w)$$
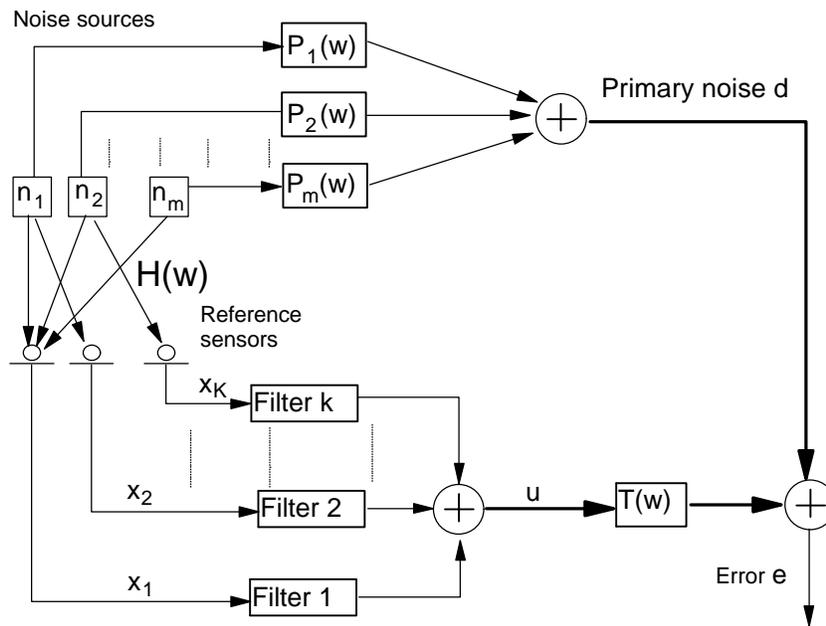
Figure 3.1 Block diagram of a single channel multiple reference active noise control system.

$$= d(w) + \sum_{i=1}^{K} T(w) x_i(w) W_i(w) \tag{3.1.3}$$

Defining the filtered reference signal as
$$\hat{x}_i(w) = x_i(w) T(w)$$

equation (3.1.3) is further simplified as
$$e(w) = d(w) + \sum_{i=1}^{K} \hat{x}_i(w) W_i(w)$$
$$= d(w) + \hat{\mathbf{x}}^T(w) \mathbf{W}(w) \tag{3.1.4}$$

where
$$\hat{\mathbf{x}}(w) = \left\{ \hat{x}_1(w) \quad \hat{x}_2(w) \quad \cdots \quad \hat{x}_k(w) \right\}^T \tag{3.1.5}$$
$$\mathbf{W}(w) = \left\{ W_1(w) \quad W_2(w) \quad \cdots \quad W_k(w) \right\}^T \tag{3.1.6}$$

The system cost function is constructed as follows
$$\xi(w) = \mathrm{E}\big(e^*(w) e(w)\big)$$
$$= \mathrm{E}\big(d^*(w) d(w) + d^*(w) \hat{\mathbf{x}}^T(w) \mathbf{W}(w) + \hat{\mathbf{x}}^H(w) \mathbf{W}^*(w) d(w)\big)$$
$$+ \mathrm{E}\big(\mathbf{W}^H(w) \hat{\mathbf{x}}^*(w) \hat{\mathbf{x}}^T(w) \mathbf{W}(w)\big) \tag{3.1.7}$$

It is important to note that the above cost function has *Hermitian quadratic form* [6], and is a real scalar. The gradient of the cost function with respect to the complex weight vector is obtained as
$$\nabla = \mathrm{E}\big(\hat{\mathbf{x}}^*(w) \hat{\mathbf{x}}^T(w)\big) \mathbf{W}(w) + \mathrm{E}\big(d(w) \hat{\mathbf{x}}^*(w)\big)$$
$$= \mathbf{R}(w) \mathbf{W}(w) + \mathbf{P}(w) \tag{3.1.8}$$

where
$$\mathbf{R}(w) = \mathbf{S}_{\hat{x}\hat{x}}(w) = \begin{pmatrix} S_{\hat{x}_1 \hat{x}_1}(w) & S_{\hat{x}_1 \hat{x}_2}(w) & \cdots & S_{\hat{x}_1 \hat{x}_k}(w) \\ S_{\hat{x}_2 \hat{x}_1}(w) & S_{\hat{x}_2 \hat{x}_2}(w) & \cdots & S_{\hat{x}_2 \hat{x}_k}(w) \\ \vdots & \vdots & \vdots & \vdots \\ S_{\hat{x}_k \hat{x}_1}(w) & S_{\hat{x}_k \hat{x}_2}(w) & \cdots & S_{\hat{x}_k \hat{x}_k}(w) \end{pmatrix} \tag{3.1.9}$$

and
$$\mathbf{P}(w) = \mathbf{S}_{\hat{x}d}(w) = \begin{pmatrix} S_{\hat{x}_1 d}(w) \\ S_{\hat{x}_2 d}(w) \\ \vdots \\ S_{\hat{x}_k d}(w) \end{pmatrix} \tag{3.1.10}$$

Since the correlation matrix $\mathbf{R}(w)$ is positive definite, the cost function $\xi(w)$ has a unique global minimum. The corresponding optimum weight vector is obtained by setting the complex gradient vector to zero, thus
$$\mathbf{W}_{opt}(w) = -\mathbf{R}^{-1}(w) \mathbf{P}(w)$$
$$= -\mathbf{S}_{\hat{x}\hat{x}}^{-1}(w) \mathbf{S}_{\hat{x}d}(w) \tag{3.1.11}$$

It is interesting to note that if the reference signals are uncorrelated, their cross-spectra are zero, it follows that all the off-diagonal terms inside the matrix $\mathbf{R}(w)$ are zero. Thus, equation (3.1.11) can be rewritten as

$$W_{i,opt}(w) = \frac{S_{\hat{x}_i d}(w)}{S_{\hat{x}_i \hat{x}_i}(w)} \qquad\qquad i = 1,2,3,\cdots, \mathrm{K} \qquad\qquad (3.1.12)$$

The above equation indicates that, if the reference signals are uncorrelated, the optimum solutions of all the filters are independent of each other, and each filter operates without any interference from other filters. In other words, the filters are uncoupled if the reference signals are uncorrelated. The optimum controller vector can be derived in terms of system transfer functions. The primary noise at the error sensors is formed by M noise sources passing through M primary paths and is given by

$$d(w) = \mathbf{n}^T(w)\mathbf{P}(w) \qquad\qquad (3.1.13)$$

where $\mathbf{n}(w)$ and $\mathbf{P}(w)$ are noise source vector and primary transfer function vector respectively and can be expressed as

$$\mathbf{n}(w) = \left\{ n_1(w) \quad n_2(w) \quad \cdots \quad n_m(w) \right\}^T \qquad\qquad (3.1.14)$$

$$\mathbf{P}(w) = \left\{ P_1(w) \quad P_2(w) \quad \cdots \quad P_m(w) \right\}^T \qquad\qquad (3.1.15)$$

The noise sources and the reference signals are related by

$$\mathbf{x}(w) = \mathbf{H}(w)\mathbf{n}(w) \qquad\qquad (3.1.16)$$

where $\mathbf{H}(w)$ is the source coupling matrix, whose element $H_{ij}(w)$ represents the transfer function between the ith noise source and the jth reference sensor, and can be written as

$$\mathbf{H}(w) = \begin{pmatrix} H_{11}(w) & H_{21}(w) & \cdots & S_{m1}(w) \\ H_{21}(w) & H_{22}(w) & \cdots & S_{m2}(w) \\ \vdots & \vdots & \vdots & \vdots \\ H_{k1}(w) & H_{k2}(w) & \cdots & S_{mk}(w) \end{pmatrix} \qquad\qquad (3.1.17)$$

Substituting equations (3.1.13), and (3.1.16) into equation(3.1.8) and moving all the transfer functions to the outside of expectation, the gradient of the cost function is obtained as

$$\nabla = T^*(w)\mathbf{H}^*(w)\mathrm{E}\big(\mathbf{n}^*(w)\mathbf{n}^T(w)\big)\big[\mathbf{H}^T(w)T(w)\mathbf{W}(w) + \mathbf{P}(w)\big] \qquad\qquad (3.1.18)$$

Another form of optimum controller vector can be obtained by setting the gradient to zero, i.e.

$$\mathbf{W}_{opt}(w) = -\mathbf{H}^+(w)\frac{\mathbf{P}(w)}{T(w)} \qquad\qquad (3.1.19)$$

where $\mathbf{H}^+(w)$ is the pseudo-inverse of $\mathbf{H}^T(w)$. The optimum controller vector is expressed in terms of transfer functions of the primary path, the error path and the source coupling path. This solution is intuitively clear: the error signal comes from the noise sources through both the primary path and the secondary path, the controller in the secondary path adjusts its transfer function, so that the two paths have the same magnitude response and 180 degrees phase difference, thus achieving noise cancellation.

3.1.2 Coherence Analysis

The cost function in equation (3.1.7) can be expressed in terms of the auto-spectrum and the cross-spectrum of the reference signal and the error signal. Equation (3.1.7) can be rewritten as

$$S_{ee}(w) = S_{dd}(w) + \mathbf{S}^T_{d\hat{x}}(w)\mathbf{W}(w) + \mathbf{S}^T_{\hat{x}d}(w)\mathbf{W}^*(w) + \mathbf{W}^H(w)\mathbf{S}_{\hat{x}\hat{x}}(w)\mathbf{W}(w) \quad (3.1.20)$$

when the filter $\mathbf{W}(w)$ is optimized using equation (3.1.11), the auto-spectrum of the error signal reaches its minimum value. Substituting the equation (3.1.11) into (3.1.20), the minimum error is obtained as

$$S_{ee}(w) = S_{dd}(w) + \mathbf{S}^T_{d\hat{x}}(w)\mathbf{S}_{\hat{x}\hat{x}}^{-1}(w)\mathbf{S}_{\hat{x}d}(w) \quad (3.1.21)$$

Dividing both sides of equation (3.1.21) by $S_{dd}(w)$ results in the relative reduction of the error signal, i.e.

$$\frac{S_{ee}(w)}{S_{dd}(w)} = 1 - \frac{\mathbf{S}^T_{d\hat{x}}(w)\mathbf{S}_{\hat{x}\hat{x}}^{-1}(w)\mathbf{S}_{\hat{x}d}(w)}{S_{dd}(w)}$$

$$= 1 - g^2_{\hat{x}d}(w) \quad (3.1.22)$$

where $g^2_{\hat{x}d}(w)$ is referred to as multiple coherence function. If all the reference signals are independent of each other, the off-diagonal terms of the matrix $\mathbf{S}_{\hat{x}\hat{x}}(w)$ are zero. Then, the above equation can also be written as

$$\frac{S_{ee}(w)}{S_{dd}(w)} = 1 - \frac{S_{\hat{x}_1 d}(w)S_{d\hat{x}_1}(w)}{S_{\hat{x}_1\hat{x}_1}(w)S_{dd}(w)} - \quad \cdots \quad - \frac{S_{\hat{x}_k d}(w)S_{d\hat{x}_k}(w)}{S_{\hat{x}_k\hat{x}_k}(w)S_{dd}(w)} \quad (3.1.23)$$

Defining the ordinary coherence function between the reference x_i and the primary noise d as

$$\gamma^2_{\hat{x}_i d}(w) = \frac{S_{\hat{x}_i d}(w)S_{d\hat{x}_i}(w)}{S_{\hat{x}_i\hat{x}_i}(w)S_{dd}(w)} \quad (3.1.24)$$

equation (3.1.23) is simplified to

$$\frac{S_{ee}(w)}{S_{dd}(w)} = 1 - g^2_{\hat{x}_1 d}(w) - \quad \cdots \quad g^2_{\hat{x}_k d}(w) \quad (3.1.25)$$

Comparing equation (3.1.22) with equation (3.1.25), it is easy to obtain

$$g^2_{\hat{x}d}(w) = g^2_{\hat{x}_1 d}(w) + \quad \cdots \quad g^2_{\hat{x}_k d}(w) \quad (3.1.26)$$

The above equation is intuitively appealing, as it indicates that better noise control effect can be achieved through the use of multiple reference signals in an environment with multiple noise sources, since the multiple coherence function $g^2_{\hat{x}d}(w)$ is greater than an ordinary reference coherence function $g^2_{\hat{x}_i d}(w)$. This explains the fundamental motivation for the use of multiple reference signals. It is important to note that equation (3.1.25) gives the maximum noise reduction that can be potentially achieved. In a real ANC system, due to causality and finite filter length, the actual control effect will generally be less.

We shall discuss details in the next section about the correlation among reference signals. Here, it is pointed out that the reference signals are very likely to be correlated in a MRANC

system. If the reference signals are correlated, equation (3.1.26) will not hold. An alternative equation can be obtained through the introduction of the partial coherence function. The partial coherence function between signal $\hat{x}_2$ and the primary noise d is defined as

$$g^2_{\hat{x}_2 d \cdot \hat{x}_1}(w) = \frac{S_{\hat{x}_2 d \cdot \hat{x}_1}(w) S_{d\hat{x}_2 \cdot \hat{x}_1}(w)}{S_{\hat{x}_2 \hat{x}_2 \cdot \hat{x}_1}(w) S_{dd \cdot \hat{x}_1}(w)} \tag{3.1.27}$$

where $S_{\hat{x}_2 d \cdot \hat{x}_1}(w)$ is referred to as the conditional spectrum, since the evaluation is based on the residual $\hat{x}_2$ and residual d, where the contribution correlated with $\hat{x}_1$ is removed from both $\hat{x}_2$ and d. Similarly, the partial coherence function between signal $\hat{x}_k$ and the primary noise d is defined as

$$g^2_{\hat{x}_k d \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w) = \frac{S_{\hat{x}_k d \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w) S_{d\hat{x}_k \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w)}{S_{\hat{x}_k \hat{x}_k \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w) S_{dd \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w)} \tag{3.1.28}$$

The evaluation of $S_{\hat{x}_k d \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w)$ is based on the residual $\hat{x}_k$ and residual d, where all the contributions correlated with $\hat{x}_1$, … and $\hat{x}_{k-1}$ are removed.

The multiple coherence function can be expressed in terms of partial coherence function as [30]

$$g^2_{\hat{x}d}(w) = 1 - \left(1 - g^2_{\hat{x}_1 d}(w)\right)\left(1 - g^2_{\hat{x}_2 d \cdot \hat{x}_1}(w)\right) \cdots \left(1 - g^2_{\hat{x}_k d \cdot \hat{x}_1 \cdots \hat{x}_{k-1}}(w)\right) \tag{3.1.29}$$

The above equation holds regardless of whether the reference signals are correlated or not. When the reference signals are uncorrelated, equation (3.1.29) reduces to equation (3.1.26).

In a MRANC system, the maximum noise reduction is bounded by the multiple coherence function. When ordinary coherence functions are obtained, care should be given using the ordinary coherence functions to evaluate the potential noise reduction. For example, consider a two reference ANC system with an ordinary coherence function between each reference signal and the primary noise of 0.5 throughout the interested frequency range. If the two reference signals are uncorrelated, then, according to equation (3.1.26), the multiple coherence function will be unity (i.e. the summation of the ordinary functions). However, if the correlation between the two reference signals is unknown, the multiple coherence may not be larger than the ordinary coherence function. In other words, the performance of MRANC may not be better than that of single reference ANC. In fact, if the reference signals are completely correlated, the multiple coherence function is still around 0.5. The actual situation can be revealed through the evaluation of the partial coherence function. When the reference signals are completely correlated, each partial coherence function is close to zero. In this case, one of the two reference signals can be essentially discarded without sacrificing any noise reduction effect.

3.1.3 Correlation Matrix and Condition Number

As discussed in the previous section, multiple reference signals are needed to achieve satisfactory noise reduction in the system where there are multiple noise sources. The reference signals are obtained through multiple sensors. Usually, each reference sensor picks up signals from several noise sources passing through different paths, as shown in Figure 3.2. As a result, the reference signals are generally correlated. The correlated part of those reference signals represents the common input to the different filters, which subsequently generates correlated outputs. Therefore, the correlation among reference signals leads to redundancy of the filter outputs. The detrimental effects are investigated in what follows:

The error signal at time step k can be expressed as
$$e(k) = d(k) + y(k)$$
$$= d(k) + u(k)T(z) \tag{3.1.30}$$
where T(z) is the Z transform of the error path, u(n) is the summation of all the filter outputs, given by

$$u(k) = \sum_{i=1}^{K} \mathbf{x}_i^{T}(k)\mathbf{w}_i = \sum_{i=1}^{K} \mathbf{w}_i^{T}\mathbf{x}_i(k) \tag{3.1.31}$$

where $\mathbf{w}_i$ and $\mathbf{x}_i(n)$ are the weight vector and the tapped input vector for the ith filter respectively. Suppose the filter length is M, then

Noise source 1

Reference sensor 1

Noise source 2

Reference sensor 2
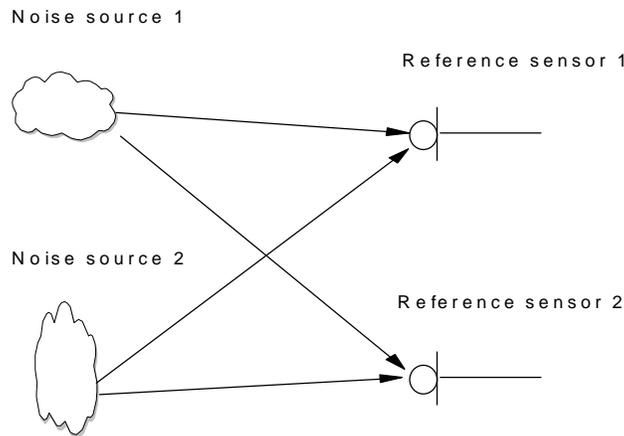
Figure 3.2 An example illustrating the correlation of the reference signals in a multiple noise source environment.

$$\mathbf{w}_i = \begin{bmatrix} w_{0,i} & w_{1,i} & w_{2,i} & \dots & w_{M-1,i} \end{bmatrix}^T \tag{3.1.32}$$

$$\mathbf{x}_i(k) = \begin{bmatrix} x_i(k) & x_i(k-1) & x_i(k-2) & \dots & x_i(k-M+1) \end{bmatrix}^T \tag{3.1.33}$$

Substituting equation (3.1.31) into equation (3.1.30) yields

$$e(k) = d(k) + \sum_{i=1}^{K} \mathbf{w}_i^T \mathbf{x}_i(k) T(z) \tag{3.1.34}$$

If the impulse response of the adaptive filter varies slowly, the error path and the adaptive filter can be commuted. The filtered reference signal is defined as

$$\hat{\mathbf{x}}_i(k) = \mathbf{x}_i(k) T(z) \tag{3.1.35}$$

Substituting equation (3.1.35) into (3.1.34), we obtain

$$e(k) = d(k) + \sum_{i=1}^{K} \mathbf{w}_i^T \hat{\mathbf{x}}_i(k)$$

$$= d(k) + \mathbf{W}^T \hat{\mathbf{X}}(k) \tag{3.1.36}$$

where $\mathbf{W}$ and $\hat{\mathbf{X}}(k)$ are vectors with K×M coefficients, i.e.

$$\mathbf{W} = \begin{pmatrix} \mathbf{w}^T{}_1 & \mathbf{w}_2{}^T & \cdots & \mathbf{w}_k{}^T \end{pmatrix}^T \tag{3.1.37}$$

$$\hat{\mathbf{X}}(k) = \begin{pmatrix} \hat{\mathbf{x}}^T{}_1(k) & \hat{\mathbf{x}}^T{}_2(k) & \cdots & \hat{\mathbf{x}}^T{}_k(k) \end{pmatrix}^T \tag{3.1.38}$$

The cost function is constructed as the mean square error signal, i.e.

$$\mathbf{x} = \mathrm{E}\!\left[e^2(k) \right] \tag{3.1.39}$$

Taking the derivative of the cost function with respect to weight vector results in

$$\nabla = \frac{\P \mathbf{x}}{\P \mathbf{W}} = 2\mathrm{E}\!\left[e(k) \frac{\P e(k)}{\P \mathbf{W}} \right]$$

$$= 2E\!\left[d(k) \hat{\mathbf{X}}(k) \right] + 2E\!\left[\hat{\mathbf{X}}(k) \hat{\mathbf{X}}^T(k) \right] \mathbf{W} \tag{3.1.40}$$

The optimum Wiener weight vector is obtained by setting the gradient of the cost function to zero, which gives

$$E\!\left[d(k) \hat{\mathbf{X}}(k) \right] + E\!\left[\hat{\mathbf{X}}(k) \hat{\mathbf{X}}^T(k) \right] \mathbf{W} = 0 \tag{3.1.41}$$

defining

$$\mathbf{P} = E\!\left[d(k) \hat{\mathbf{X}}(k) \right]$$

and

$$\mathbf{R} = E\!\left[\hat{\mathbf{X}}(k) \hat{\mathbf{X}}^T(k) \right]$$

equation (3.1.41) can be simplified as

$$\mathbf{RW} = -\mathbf{P} \tag{3.1.42}$$

the above equation is called the *norm equation*, its characteristic is largely determined by the correlation matrix $\mathbf{R}$, which can be expanded as

$$\mathbf{R} = E\begin{pmatrix} \hat{\mathbf{x}}_1(k)\hat{\mathbf{x}}^T_1(k) & \hat{\mathbf{x}}_1(k)\hat{\mathbf{x}}^T_2(k) & \cdots & \hat{\mathbf{x}}_1(k)\hat{\mathbf{x}}^T_k(k) \\ \hat{\mathbf{x}}_2(k)\hat{\mathbf{x}}^T_1(k) & \hat{\mathbf{x}}_2(k)\hat{\mathbf{x}}^T_2(k) & \cdots & \hat{\mathbf{x}}_2(k)\hat{\mathbf{x}}^T_k(k) \\ \vdots & \vdots & \vdots & \vdots \\ \hat{\mathbf{x}}_k(k)\hat{\mathbf{x}}^T_1(k) & \hat{\mathbf{x}}_k(k)\hat{\mathbf{x}}^T_2(k) & \cdots & \hat{\mathbf{x}}_k(k)\hat{\mathbf{x}}^T_k(k) \end{pmatrix} \tag{3.1.43}$$

It is important to note that each term inside the above matrix is a sub-matrix, and the whole matrix is generally not *Toepolitz* [6], as contrasted to the auto-correlation matrix of a single reference ANC system. A square matrix is Toepolitz if all the elements on the main diagonal are equal and any other elements on the subordinating diagonal are also equal. On the other hand, the matrix $\mathbf{R}$ is real, symmetric and non-negative definite just like the matrix $\mathbf{R}$ in a single reference ANC system. Therefore, the corresponding eigenvalues are also non-negative and real. The characteristics of the matrix $\mathbf{R}$ are determined by the cross-correlation functions of the reference signals and the auto-correlation functions of the reference signals. If all the reference signals are uncorrelated with each other, every off-diagonal terms in the matrix $\mathbf{R}$ will be zero and every optimum weight vectors, $\mathbf{w}_1$, $\mathbf{w}_2$, ... and $\mathbf{w}_k$, are uncoupled.

It is very important to understand *perturbation theory* [31] and its impact on the development of an algorithm to solve the norm equation (3.1.42). The perturbation theory states that if the matrix $\mathbf{R}$ and the vector $\mathbf{P}$ are perturbed by small amounts $\delta\mathbf{R}$ and $\delta\mathbf{P}$ respectively, and if the relative perturbations, $||\delta\mathbf{R}||/||\mathbf{R}||$ and $||\delta\mathbf{P}||/||\mathbf{P}||$, are both on the same order of ε, where $\varepsilon \ll 1$, then

$$\frac{\|d\,\mathbf{W}\|}{\|\mathbf{W}\|} \leq e\,c(\mathbf{R}) \tag{3.1.44}$$

where $\delta\mathbf{W}$ is the change of weight vector $\mathbf{W}$ as a result of the perturbation from the matrix $\mathbf{R}$ and the vector $\mathbf{P}$, and $\chi(\mathbf{R})$ is the *condition number* of the matrix $\mathbf{R}$, and $\|\cdot\|$ is the *norm* operator [32]. The condition number describes the ill condition of a matrix. Since the matrix R is real and symmetric, it can be shown [6] that the condition number equals

$$c(\mathbf{R}) = \frac{l_{max}}{l_{min}} \tag{3.1.45}$$

where l_{max} and l_{min} are the maximum and minimum eigenvalues of the matrix $\mathbf{R}$ respectively. This ratio is also commonly referred to as eigenvalue spread.

The perturbation theory states that if there are some errors in the matrix $\mathbf{R}$ or the vector $\mathbf{P}$ caused by measurement noises or some other reasons, the ill-condition of the correlation matrix $\mathbf{R}$ may lead to a weight vector solution $\mathbf{W}$ which is far from the optimum Wiener solution $\mathbf{W}_{opt}$. In other words, the ill-conditioning of the correlation matrix $\mathbf{R}$ causes the optimum Wiener vector to be very sensitive to measurement errors.

On the other hand, as discussed in section 2.3.1, the eigenvalue spread of the matrix $\mathbf{R}$ has a significant impact on the convergence rate of an ANC system, especially when the LMS algorithm is applied. An important factor which determines the eigenvalue spread is the correlation among the reference signals. In particular, if the reference signal x_i is correlated with

the reference signal x_j, the ith and jth columns in the matrix **R** will exhibit some similarities which result in large eigenvalue spread, and accordingly slows down the convergence speed. Upon the extreme circumstance when any two reference signals are the same, the determinant of the matrix R becomes zero, and the eigenvalue spread goes up to infinite.

Two simple systems are now considered as shown in Fig. 3.4a. and Fig. 3.4b, in which there are only two reference signals in each system. Suppose each filter in the systems has only two weights, then the correlation matrix R becomes

$$R = E\begin{pmatrix} \hat{x}_1(k)\hat{x}_1(k) & \hat{x}_1(k)\hat{x}_1(k-1) & \hat{x}_1(k)\hat{x}_2(k) & \hat{x}_1(k)\hat{x}_2(k-1) \\ \hat{x}_1(k-1)\hat{x}_1(k) & \hat{x}_1(k-1)\hat{x}_1(k-1) & \hat{x}_1(k-1)\hat{x}_2(k) & \hat{x}_1(k-1)\hat{x}_2(k-1) \\ \hat{x}_2(k)\hat{x}_1(k) & \hat{x}_2(k)\hat{x}_1(k-1) & \hat{x}_2(k)\hat{x}_2(k) & \hat{x}_2(k)\hat{x}_2(k-1) \\ \hat{x}_2(k-1)\hat{x}_1(k) & \hat{x}_2(k-1)\hat{x}_1(k-1) & \hat{x}_2(k-1)\hat{x}_2(k) & \hat{x}_2(k-1)\hat{x}_2(k-1) \end{pmatrix} \quad (3.1.46)$$

It is important to note that the terms inside the above correlation matrix **R** is the filtered reference signal instead of the reference signal due to the error path in an ANC system. The first case is shown in Figure 3.3a. The first filtered reference signal $\hat{x}_1$ is a sine wave with the frequency ω_1, and the second filtered reference signal $\hat{x}_2$ is a sine wave with the frequency ω_2 corrupted by some contribution from $\hat{x}_1$. The two filtered reference signal is written as

$$\hat{x}_1 = a_1 \sin(w_1 t) \quad (3.1.47a)$$
$$\hat{x}_2 = a_2 \sin(w_2 t) + e \sin(w_1 t) \quad (3.1.47b)$$

Since the two signals $\hat{x}_1$ and $\hat{x}_2$ are deterministic, the correlation function between them can be analytically evaluated as

$$R_{x_1 x_1}(t) = E\big(x_1(t)x_1(t+t)\big)$$
$$= E\big(a_1 \sin(w_1 t)a_1 \sin(w_1(t+t))\big)$$
$$= E\left(\frac{1}{2}a_1^2\big(\cos(w_1 t) - \cos(2w_1 t + w_1 t)\big)\right)$$
$$= \frac{1}{2}a_1^2 \cos(w_1 t) \quad (3.1.48a)$$

$$R_{x_2 x_2}(t) = E\big(x_2(t)x_2(t+t)\big)$$
$$= E\big((a_2 \sin(w_2 t) + e \sin(w_1 t))(a_2 \sin(w_2(t+t)) + e \sin(w_1(t+t)))\big)$$
$$= \frac{1}{2}a_2^2 \cos(w_2 t) + \frac{1}{2}e^2 \cos(w_1 t) \quad (3.1.48b)$$

$$R_{x_1 x_2}(t) = E\big(x_1(t)x_2(t+t)\big)$$
$$= E\big(a_1 \sin(w_1 t)(a_2 \sin(w_2(t+t)) + e \sin(w_1(t+t)))\big)$$
$$= \frac{1}{2}a_1 e \cos(w_1 t) \quad (3.1.48c)$$

Using the above equations (3.1.48), all the terms inside the correlation matrix R are given by

$\hat{x}_1 = a_1 \sin(w_1 t)$

x_1

Error Path

Filter 1

u_1

d

$\hat{x}_2 = a_2 \sin(w_2 t) + e\sin(w_1 t)$

x_2

Error Path

Filter 2

Error
e

(a)

$\hat{x}_1 = a_1 \sin(w_1 t) + n_1$

x_1

Error Path

Filter 1

u_1

d

$\hat{x}_2 = a_2 \sin(w_2 t) + n_2$

x_2

Error Path

Filter 2

Error e
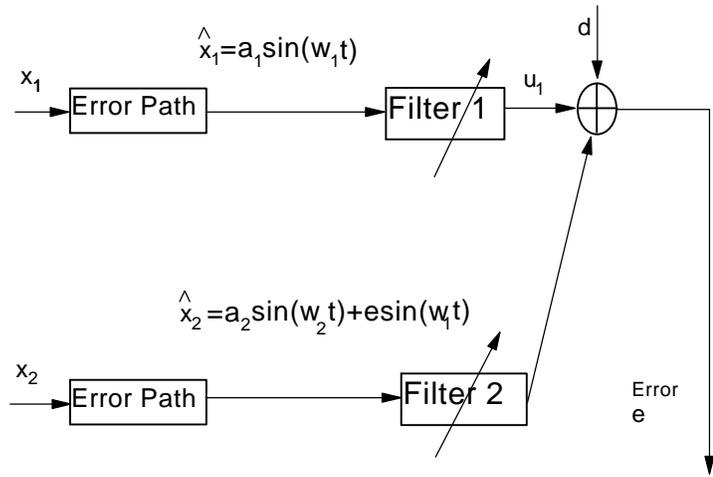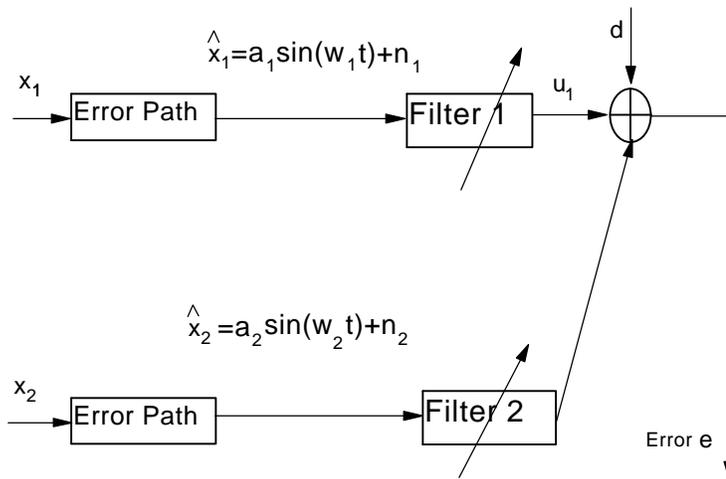
(b)

Figure 3.3 Case study of single channel MRANC systems, a) reference signals are correlated due to coupling, b) reference signals are correlated due to common noise.

$$E\left(\hat{x}_1(k)\hat{x}_1(k)\right) = \frac{1}{2}a^2_1 \qquad\qquad E\left(\hat{x}_1(k)\hat{x}_1(k-1)\right) = \frac{1}{2}a^2_1\cos\omega_1\Delta t$$

$$E\left(\hat{x}_1(k)\hat{x}_2(k)\right) = \frac{1}{2}a_1 e \qquad\qquad E\left(\hat{x}_1(k)\hat{x}_2(k-1)\right) = \frac{1}{2}a_1 e\cos\omega_1\Delta t$$

$$E\left(\hat{x}_1(k-1)\hat{x}_2(k)\right) = \frac{1}{2}a_1 e\cos\omega_1\Delta t \quad E\left(\hat{x}_2(k)\hat{x}_2(k)\right) = \frac{1}{2}a^2_2 + \frac{1}{2}e^2$$

$$E\left(\hat{x}_2(k)\hat{x}_2(k-1)\right) = \frac{1}{2}a^2_2\cos\omega_2\Delta t + \frac{1}{2}e^2\cos\omega_1\Delta t \qquad\qquad (3.1.49)$$

where Δt is the sampling interval, which is related to the sampling frequency by $\Delta t = 1/f_s$.
Substituting the above equations (3.1.49) into (3.1.46) yields

$$R = \frac{1}{2}\begin{pmatrix} a_1^2 & a_1^2\cos\omega_1\Delta t & a_1 e & a_1 e\cos\omega_1\Delta t \\ a_1^2\cos\omega_1\Delta t & a_1^2 & a_1 e\cos\omega_1\Delta t & a_1 e \\ a_1 e & a_1 e\cos\omega_1\Delta t & a_2^2 + e^2 & a_2^2\cos\omega_2\Delta t + e^2\cos\omega_1\Delta t \\ a_1 e\cos\omega_1\Delta t & a_1 e & a_2^2\cos\omega_2\Delta t + e^2\cos\omega_1\Delta t & a_2^2 + e^2 \end{pmatrix}$$

$$(3.1.50)$$

The correlation between the two filtered reference signals can be modified by setting the parameter e to different values. The eigenvalue spread $\lambda_{max}/\lambda_{min}$ is calculated for different e. The results are shown in Table 3.1. It is clear that the eigenvalue spread goes up when the correlation between the two filtered reference signals increases. Upon the extreme case when the two filtered reference signals have the same frequency (a_2=0, $e\neq$0), which means that one signal can be considered as a linear transformation of the other signal, the eigenvalue spread of the matrix **R** will be infinite, and the system becomes underdetermined. An underdetermined system results in infinite number of solutions for the norm equation (3.1.42). Among these solutions, some give rise to very large control efforts, which is generally not desirable for an ANC system.

Table 3.1 Eigenvalue spread versus correlation, case No.1

a_1=1, a_2=1, ω_1=100*2π Hz, ω_2=120*2π Hz, Δt=0.003s				
Correlation	e=0	e=0.2	e=0.5	e=0.8	e=1.0
$\lambda_{max}/\lambda_{min}$	4.5161	5.4201	9.1461	16.0819	23.2590

The second case considered is shown in Figure 3.3b. The first filtered reference signal $\hat{x}_1$ is a sine wave with the frequency ω_1 corrupted by the noise n_1, and the second filtered reference signal $\hat{x}_2$ is a sine wave with the frequency ω_2 corrupted by the noise n_2. The two filtered reference signals is written as

$$\hat{x}_1 = a_1\sin(\omega_1 t) + n_1 \qquad\qquad (3.1.51a)$$
$$\hat{x}_2 = a_2\sin(\omega_2 t) + n_2 \qquad\qquad (3.1.51b)$$

the two noise signals, n_1 and n_2, are assumed to be correlated, and their correlation can be simply modeled as

$$
f_{n_1 n_2}(k) = \begin{cases} 0.5e & k = 0 \\ \dfrac{2}{5}e & k = 1 \\ 0 & k = else \end{cases} \tag{3.1.52}
$$

where the parameter ε is a constant describing the correlation between the two noise signals. Using equation (3.1.48), all the terms inside the correlation matrix $\mathbf{R}$ can be evaluated as follows

$$
E\big(\hat{x}_1(k)\hat{x}_1(k)\big) = \frac{1}{2}a^2_1 + \frac{1}{2}e \qquad E\big(\hat{x}_1(k)\hat{x}_1(k-1)\big) = \frac{1}{2}a^2_1 \cos w_1 \Delta t + \frac{2}{5}e
$$

$$
E\big(\hat{x}_1(k)\hat{x}_2(k)\big) = \frac{1}{2}e \qquad E\big(\hat{x}_1(k)\hat{x}_2(k-1)\big) = \frac{2}{5}e
$$

$$
E\big(\hat{x}_1(k-1)\hat{x}_2(k)\big) = \frac{2}{5}e \qquad E\big(\hat{x}_2(k)\hat{x}_2(k)\big) = \frac{1}{2}a^2_2 + \frac{1}{2}e
$$

$$
E\big(\hat{x}_2(k)\hat{x}_2(k-1)\big) = \frac{1}{2}a^2_2 \cos w_2 \Delta t + \frac{2}{5}e \tag{3.1.53}
$$

Again, Δt is the sampling interval, substituting the above equations into (3.1.46) yields

$$
R = \frac{1}{2}\begin{pmatrix} a_1^2 + \varepsilon & a_1^2 \cos \omega_1 \Delta t + 0.8\varepsilon & \varepsilon & 0.8\varepsilon \\ a_1^2 \cos \omega_1 \Delta t + 0.8\varepsilon & a_1^2 + \varepsilon & 0.8\varepsilon & \varepsilon \\ \varepsilon & 0.8\varepsilon & a_2^2 + \varepsilon & a_2^2 \cos \omega_2 \Delta t + 0.8\varepsilon \\ 0.8\varepsilon & \varepsilon & a_2^2 \cos \omega_2 \Delta t + 0.8\varepsilon & a_2^2 + \varepsilon \end{pmatrix}
$$

$$
\tag{3.1.54}
$$

The correlation between the two filtered reference signals can be modified by setting the parameter ε to different values. The eigenvalue spread $\lambda_{max}/\lambda_{min}$ is again calculated for different ε. The results are shown in the Table 3.2. It is clear that the eigenvalue spread goes up when the correlation between the two filtered reference signals increases. Here, the correlation is caused by some correlated noises. It is important to note that the correlated noises in the reference signals will generally stretch the eigenvalue spread. As a result, the convergence rate slows down and the system performance deteriorates.

Table 3.2 Eigenvalue spread versus correlation, case No.2

$a_1=1$, $a_2=1$, $\omega_1=100*2\pi$, $\omega_2=160*2\pi$, $\Delta t=0.002$					
Correlation	$\varepsilon=0$	$\varepsilon=0.2$	$\varepsilon=0.5$	$\varepsilon=0.8$	$\varepsilon=1.0$
$\lambda_{max}/\lambda_{min}$	2.4830	2.4917	3.6184	4.7303	5.4504

3.1.4 Preprocessing of Reference Signals with Decorrelation Filters.

As discussed in the last section, the correlation between the reference signals leads to the problem of slow convergence rate and high sensitivity to measurement error. Thus, in order to improve the performance of a multiple reference control system, it is desirable to process the reference signals in some way such that they are uncorrelated with each other. In this thesis, this is achieved through a set of decorrelation filters shown in Figure 3.4. The corresponding ANC system with decorrelation filters to preprocess the reference signals is shown in Figure 3.6. We first derive the statistical relationship between the reference signal and the error signal in an ANC system shown in Figure 2.3. Recall from equation (2.2.5) that

$$e(k) = d(k) + \mathbf{x}^T(k)\mathbf{w}T(z) \tag{3.1.55}$$

Setting T(z) to unity and multiplying both sides of the equation by $\mathbf{x}$(k) yield

$$\mathbf{x}(k)e(k) = \mathbf{x}(k)d(k) + \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w} \tag{3.1.56}$$

Taking the expected value of the above equation, we obtain

$$E(\mathbf{x}(k)e(k)) = \mathbf{P} + \mathbf{R}\mathbf{w} \tag{3.1.57}$$

If the adaptive weight vector $\mathbf{w}$ is converged to the optimum Wiener vector, i.e.
$\mathbf{w} = \mathbf{w}^* = -\mathbf{R}^{-1}\mathbf{P}$ as shown in equation (1.1.13), we get

$$E(\mathbf{x}(k)e(k)) = 0 \tag{3.1.58}$$

The above equation states the *principle of orthogonality*; when the filter operates in its optimum condition, the error signal is uncorrelated with (orthogonal to) the reference signal.

Based on the orthogonal theorem, adaptive decorrelation filters can be constructed as shown in Fig. 5, in which two correlated reference signals are processed by a couple of adaptive filters to generate two uncorrelated reference signals. For the upper filter $\mathbf{A}$, the reference signal is $\bar{x}_1$, and the error signal is $\bar{x}_2$. Thus, the orthogonal relationship is expressed as

$$E\left[\bar{x}_1(k-i)\bar{x}_2(k)\right]_{\mathbf{A}=\mathbf{A}_{opt}} = 0 \qquad i = 0, 1, 2, \cdots M-1 \tag{3.1.59}$$

where M is the number of filter coefficients corresponding to filter $\mathbf{A}$. For the lower filter $\mathbf{B}$, the reference signal is $\bar{x}_2$, and the error signal is $\bar{x}_1$. Thus, the orthogonal relationship is expressed as

$$E\left[\bar{x}_2(k-i)\bar{x}_1(k)\right]_{\mathbf{B}=\mathbf{B}_{opt}} = 0 \qquad i = 0, 1, 2, \cdots M-1 \tag{3.1.60}$$

where M is the number of filter coefficients corresponding to filter $\mathbf{B}$. It is important to note that the degree of decorrelation between the reference signal and the error signal depends on the number of filter coefficients. Ideally, infinite number of filter coefficients is needed to decorrelate the two reference signals. However, since the objective of applying decorrelation filters is to improve the condition number of the matrix $\mathbf{R}$ or diagonalize it. It is ready to obtain that the two transversal filters in the decorrelation filters should have the same number of coefficients, and the number of coefficients M should be the same as the number of the controller filter coefficients.

It should be noted that the first coefficients a_0 and b_0 in the filters $\mathbf{A}$ and $\mathbf{B}$ are redundant, since both of them are trying to achieve

$$\mathrm{E}\left[\bar{x}_2(k)\bar{x}_1(k)\right]=0 \tag{3.1.61}$$

This is an over-determined case, which gives infinite solutions to a_0 and b_0. A practical approach is to force either a_0 or b_0 to be zero. Using the LMS algorithm and assuming that first coefficient b_0 is set to be zero, the uncorrelated reference signals $\bar{x}_1$ and $\bar{x}_2$ can be obtained as

$$\bar{x}_1(k) = x_1(k) + \sum_{i=1}^{M}\bar{x}_2(k-i)b_i(k) \qquad i=1,\,2,\,\cdots M-1 \tag{3.1.62a}$$

$$b_i(k+1) = b_i(k) - m\bar{x}_2(k-i)\bar{x}_1(k) \qquad i=1,\,2,\,\cdots M-1 \tag{3.1.62b}$$

$$\bar{x}_2(k) = x_2(k) + \sum_{i=0}^{M}\bar{x}_1(k-i)a_i(k) \qquad i=0,\,1,\,2,\,\cdots M-1 \tag{3.1.62c}$$

$$a_i(k+1) = a_i(k) - m\bar{x}_1(k-i)\bar{x}_2(k) \qquad i=0,\,1,\,2,\,\cdots M-1 \tag{3.1.62d}$$

This decorrelation technique has been reported to achieve signal separation or the restoration of original signals [33][34]. It is important to note that all the essential information in the reference signals is retained through the decorrelation filters. By decorrelating all the reference signals, only redundant information is left out. The decorrelation filters can be implemented using either Wiener filters or adaptive filters. If the computational load is a major concern for a MRANC system and the correlation among reference signals remain unchanged, Wiener filters should be implemented, otherwise, implementation with adaptive filters is highly suggested. The adaptive nature of the decorrelation filters can be applied to a real system more readily and effectively since there are no elaborate procedures needed to analyze the exact properties of the reference signals. Furthermore, the adaptive filters can automatically track the changing relationship among the reference signals.

If the decorrelation filters are applied without any prior knowledge of the quality of the reference signals, it is possible that one output signal becomes very small relative to the corresponding input reference signal. In this case, a conclusion can be drawn that the particular reference is highly correlated with other reference signals, and therefore this reference signal may be eliminated without compromising much noise reduction effect. It is therefore suggested to analyze the reference signals before applying the decorrelation filters. In fact, the decorrelation filters can also be used to optimize the locations of the reference sensors. If the reference signals are completely uncorrelated with each other, the output signals of the decorrelation filters would be unchanged. In real systems, it is often impossible to get complete uncorrelated reference signals, and the correlation among them is largely determined by the position of the reference sensors. The criterion for optimizing the positions of the reference sensors is then to make the output value of the decorrelation filters close in magnitude to the corresponding input value.
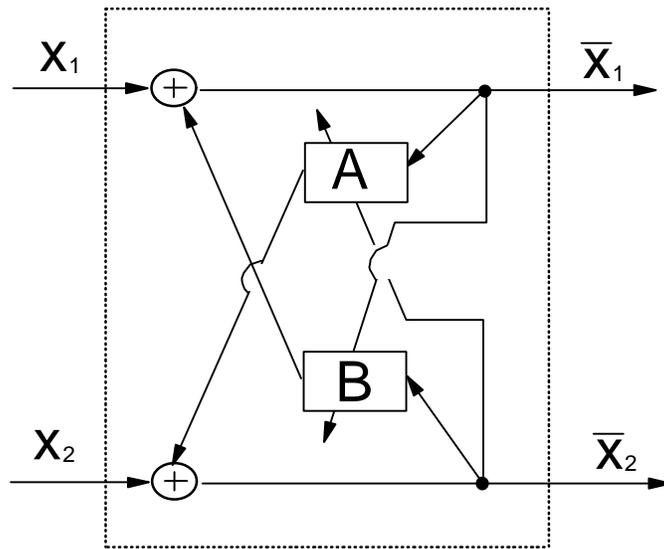
Figure 3.4 Adaptive decorrelation filters for two reference signals.
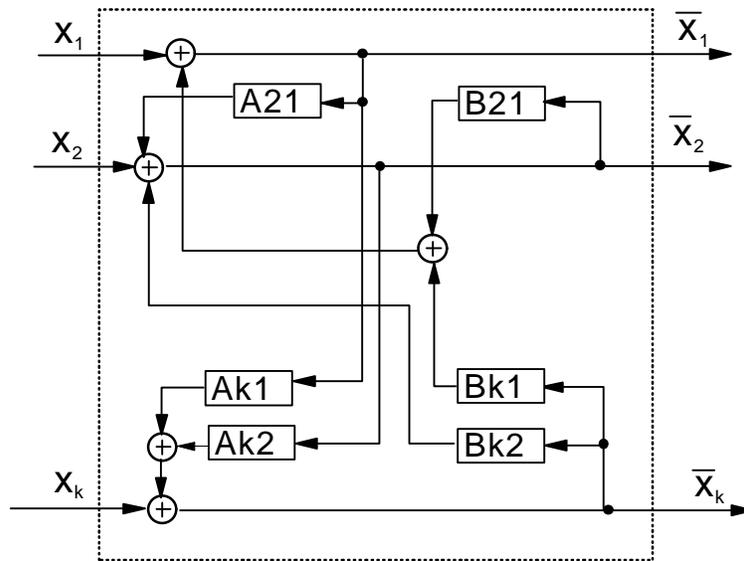
Figure 3.5 Adaptive decorrelation filters for K reference signals.
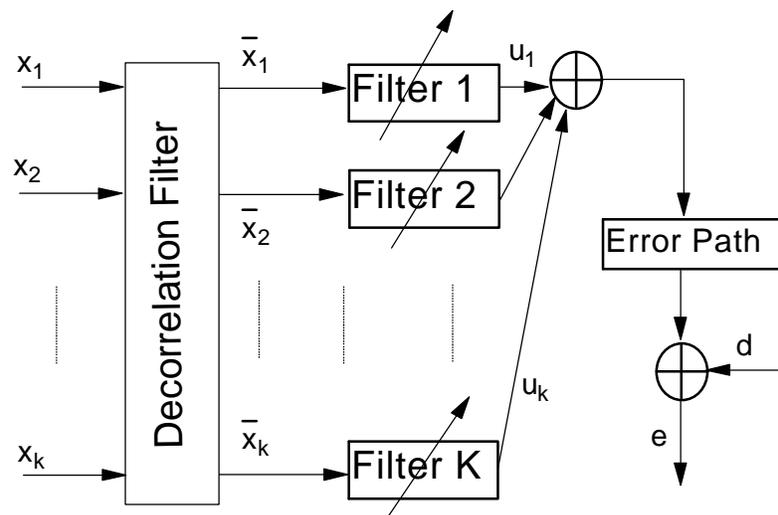
Figure 3.6 Block diagram of a single channel multiple reference ANC system, reference signals are preprocessed by decorrelation filters.

3.1.5 Multiple Reference Single Input(MRSI) and Multiple Reference Multiple Input(MRMI)

The foregoing sections discuss a multiple reference multiple input (MRMI) system, which feeds each reference signal into a different filter. One might consider forming a single input by summing all the reference signals together. Such a system is referred to as a multiple reference single input (MRSI) system, as shown in Fig. 3.6. In general, when there are multiple noise sources, a MRSI system does not perform as well as a MRMI system. This can be corroborated through coherence analysis.

As pointed out in the previous section, the maximum noise reduction effect is influenced by the coherence function between the reference signal and the desired signal. Assuming that all the reference signals x_1, x_2, x_3, ..., x_k are uncorrelated and have zero mean values, i.e.

$$u(k) = \mathrm{E}\Big(x_i(n+k)x_j(n)\Big) = 0 \qquad\qquad i \neq j \qquad\qquad (3.1.63)$$

Since the reference signals satisfy the above equation, they are also referred to as orthogonal. The cross-spectrum of any two reference signals is the discrete Fourier transform of the corresponding cross-correlation function. Since the reference signals are uncorrelated, their corresponding Fourier transforms is zero. Thus

$$S_{x_i x_j}(w) = \lim_{N \to \infty} \frac{1}{N} \mathrm{E}\Big(X_{i,N}(w)X^*_{j,N}(w)\Big) \qquad\qquad i \neq j$$

$$= \sum_{k=-\infty}^{\infty} u(k)e^{-jwk}$$

$$= 0 \qquad\qquad\qquad\qquad\qquad\qquad (3.1.64)$$

where N is the window length. The coherence function is defined as

$$g(w) = \frac{S_{xd}(w)S_{dx}(w)}{S_{xx}(w)S_{dd}(w)} \qquad\qquad (3.1.65)$$

According to Fig. 3.7, the reference signal X(w) and the desired signal D(w) can be written as

$$X(w) = X_1(w) + X_2(w) + \cdots + X_k(w) \qquad\qquad (3.1.66)$$

$$D(w) = X_1(w)H_1(w) + X_2(w)H_2(w) + \cdots + X_k(w)H_k(w) \qquad\qquad (3.1.67)$$

Taking the conjugation on both sides of equation (3.1.66) and (3.1.67) yields

$$X^*(w) = X^*_1(w) + X^*_2(w) + \cdots + X^*_k(w) \qquad\qquad (3.1.68)$$

$$D^*(w) = X^*_1(w)H^*_1(w) + X^*_2(w)H^*_2(w) + \cdots + X^*_k(w)H^*_k(w) \qquad\qquad (3.1.69)$$

Since the cross-spectrum of any two reference signals is zero, it is ready to obtain that

$$S_{xd}(w) = S_{x_1 x_1}(w)H_1(w) + S_{x_2 x_2}(w)H_2(w) + \cdots + S_{x_k x_k}(w)H_k(w) \qquad\qquad (3.1.70)$$

$$S_{dx}(w) = S_{x_1 x_1}(w)H^*_1(w) + S_{x_2 x_2}(w)H^*_2(w) + \cdots + S_{x_k x_k}(w)H^*_k(w) \qquad\qquad (3.1.71)$$

$$S_{xx}(w) = S_{x_1 x_1}(w) + S_{x_2 x_2}(w) + \cdots + S_{x_k x_k}(w) \qquad\qquad (3.1.72)$$

$$S_{dd}(w) = S_{x_1 x_1}(w)|H_1(w)|^2 + S_{x_2 x_2}(w)|H_2(w)|^2 + \cdots + S_{x_k x_k}(w)|H_k(w)|^2 \qquad\qquad (3.1.73)$$

Substituting equation (3.1.70), (3.1.71), (3.1.72) and (3.1.73) into (3.1.65) results in

$$g(w) = \frac{\sum\limits_{i=1}^{k} S_{x_i x_i}(w) H_i(w) \sum\limits_{j=1}^{k} S_{x_j x_j}(w) H^*_j(w)}{\sum\limits_{i=1}^{k} S_{x_i x_i}(w) \sum\limits_{j=1}^{k} S_{x_j x_j}(w) \left| H_j(w) \right|^2}$$

$$\leq 1 \tag{3.1.74}$$

It can be proved that the above coherence function is usually less than unity if $H_1(w)$, $H_2(w)$, ... and $H_k(w)$ are not equal. Subtracting the numerator from the denominator and omitting w in the spectral expression yield

$$S_{xx} S_{dd} - S_{xd} S_{dx}$$

$$= \sum_{i,j=1,i\neq j}^{k} \left(S^2_{x_i x_i} |H_i(w)|^2 + S_{x_i x_i} S_{x_j x_j} \left(|H_i(w)|^2 + |H_j(w)|^2 \right) + S^2_{x_j x_j} |H_j(w)|^2 \right)$$

$$- \sum_{i,j=1,i\neq j}^{k} \left(S^2_{x_i x_i} |H_i(w)|^2 + S_{x_i x_i} S_{x_j x_j} \left(H_i(w) H^*_j(w) + H^*_i(w) H_j(w) \right) + S^2_{x_j x_j} |H_j(w)|^2 \right)$$

$$= \sum_{i,j=1,i\neq j}^{k} \left(S_{x_i x_i} S_{x_j x_j} \left(|H_i(w)|^2 + |H_j(w)|^2 - H_i(w) H^*_j(w) - H^*_i(w) H_j(w) \right) \right)$$

$$= \sum_{i,j=1,i\neq j}^{k} \left(S_{x_i x_i} S_{x_j x_j} |H_i(w) - H_j(w)|^2 \right) \tag{3.1.75}$$

Under two special circumstances: 1) when all the transfer functions of the primary paths are equal, i.e. $|H_i(w) - H_j(w)| = 0 \ \ for \ i \neq j$, 2) when each reference signal occupies different frequency range, i.e. $S_{x_i x_i}(w) S_{x_j x_j}(w) = 0 \ \ \ for \ i \neq j$, equation (3.1.75) is equal to zero, which implies unity coherence function throughout the frequency range. In this case, the performance of a MRSI configuration is expected to be equivalent to that of a MRMI configuration. In fact, If the noise sources are adjacently located, and the error sensors are far away from the noise sources, all the primary paths are expected to be comparable. As a result, a similar noise reduction effect can be achieved with a MRSI configuration. The choice of a MRSI configuration greatly simplifies an multiple reference ANC system as compared to a MRMI configuration.

Since the denominator is usually larger than the numerator as shown in equation (3.1.75), the coherence function is usually less than unity. It should be noted that there is no uncontrollable extra noise in the system shown in Fig. 3.6. However, since the coherence function is less than unity, perfect noise cancellation can not be achieved at the error sensor. In an environment with multiple noise sources, the signal at the error sensor is due to multiple noise sources passing through multiple primary paths. Such an environment can be modeled perfectly only through a multiple input multiple output system. However, since the controller in the MRSI system has only a single input and a single output, it is not able to cancel the noise perfectly at the error sensor. The optimum solution for the FIR filter can be obtained from equation (2.2.10), which is repeated as follows

$$\mathrm{E}\left[d(k) \hat{\mathbf{x}}(k) \right] + \mathbf{w}_{opt} \mathrm{E}\left[\mathbf{x}(k) \mathbf{x}^T(k) \right] = 0 \tag{3.1.76}$$

where

$$\mathbf{x}(k) = \mathbf{x}_1(k) + \mathbf{x}_2(k) + \cdots + \mathbf{x}_k(k) \tag{3.1.77}$$

$$d(k) = d_1(k) + d_2(k) + \cdots + d_k(k) \tag{3.1.78}$$

Substituting equation (3.1.77) and (3.1.78) into (3.1.76) and considering x_1, x_2, ... and x_k are orthogonal, the above equation can be written as

$$\sum_{i=1}^{k} \mathbf{P}_i + \mathbf{w}_{opt} \sum_{i=1}^{k} \mathbf{R}_i = 0 \tag{3.1.79}$$

where $\mathbf{P}_i$ and $\mathbf{R}_i$ are expressed as

$$\mathbf{P}_i = \mathrm{E}\big(d(k)\mathbf{x}_i(k)\big)$$

$$\mathbf{R}_i = \mathrm{E}\big(\mathbf{x}_i(k)\mathbf{x}^T{}_i(k)\big)$$

It follows that the optimum weight vector is

$$\mathbf{w}_{opt} = -(\sum_{i=1}^{k} \mathbf{R}_i)^{-1} \sum_{i=1}^{k} \mathbf{P}_i \tag{3.1.80}$$

Supposing that the primary paths are modeled by FIR filters with weight vectors $\mathbf{w}_1$, $\mathbf{w}_2$ … and $\mathbf{w}_k$, the above equation (3.1.80) can be rewritten as

$$\mathbf{w}_{opt} = -(\sum_{i=1}^{k} \mathbf{R}_i)^{-1} (\sum_{i=1}^{k} \mathbf{R}_i \mathbf{w}_i) \tag{3.1.81}$$

The above equation gives the optimum weight vector in terms of the primary paths and the reference signals. Again, such a filter cannot perform as well as those filters in the MRMI system. This can be demonstrated by a simple case shown in Fig. 3.7, in which two reference signals are orthogonal with the same signal power, and each primary path is a FIR filter with a single weight. According to equation (3.1.81), the optimum weight for the single weight filter is $w_{opt} = 2.5$, and the corresponding error signal after control would be $0.5(x_2 - x_1)$. The power of the error signal after control is calculated as

$$P_e = \mathrm{E}\big(0.25(x_2 - x_1)^2\big)$$

$$= 0.25 \cdot \big(P_{x_2} + P_{x_1}\big) \tag{3.1.82}$$

The error signal stays the same in this case even if the control filter has more than one weight. On the other hand, if the two reference signals are not added up , but each signal is fed into a different filter, perfect noise cancellation can be achieved as discussed section (3.1.2).
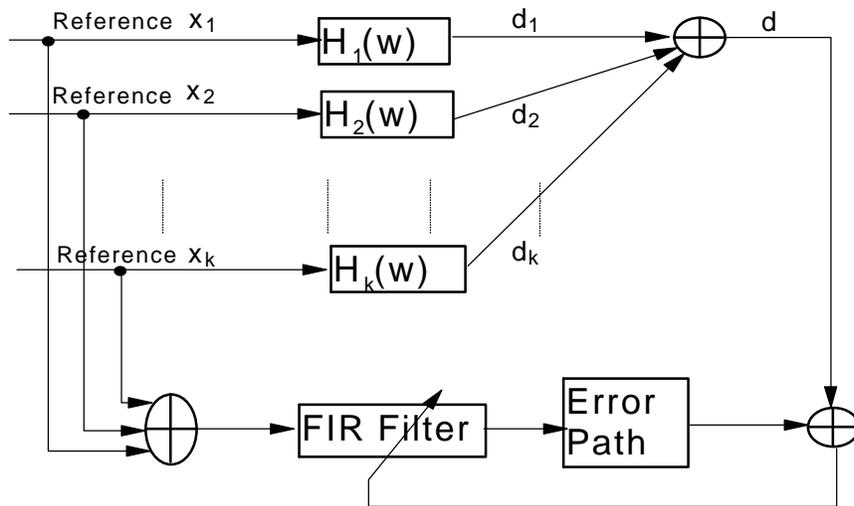
Figure 3.7 Multiple reference single input ANC system.
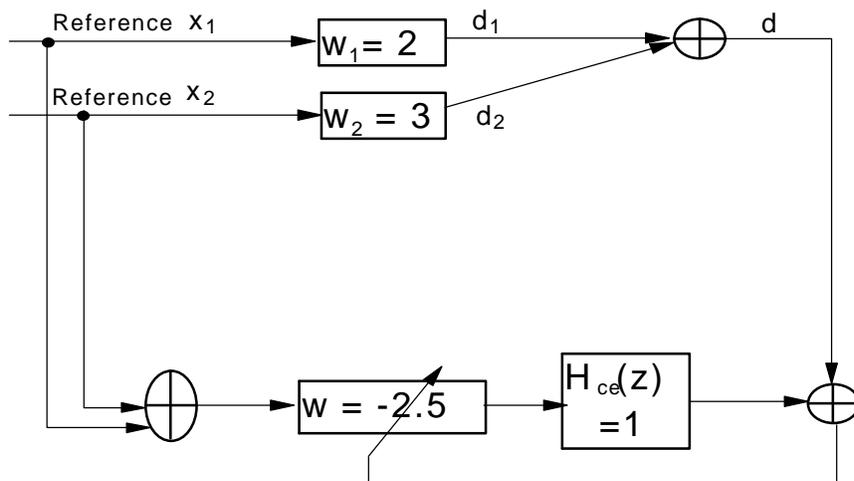
Figure 3.8 A simple case of multiple reference single input ANC system.

Reference x_1

Filter 11
Filter 12
Filter 1K

u_1

d_1 Error e_1

Reference x_2

Filter 21
Filter 21
Filter 2K

u_2

d_2 Error e_2

d_3 Error e_3

Reference x_k

Filter M1
Filter M2
Filter MK

u_M

d_L Error e_L

(a)
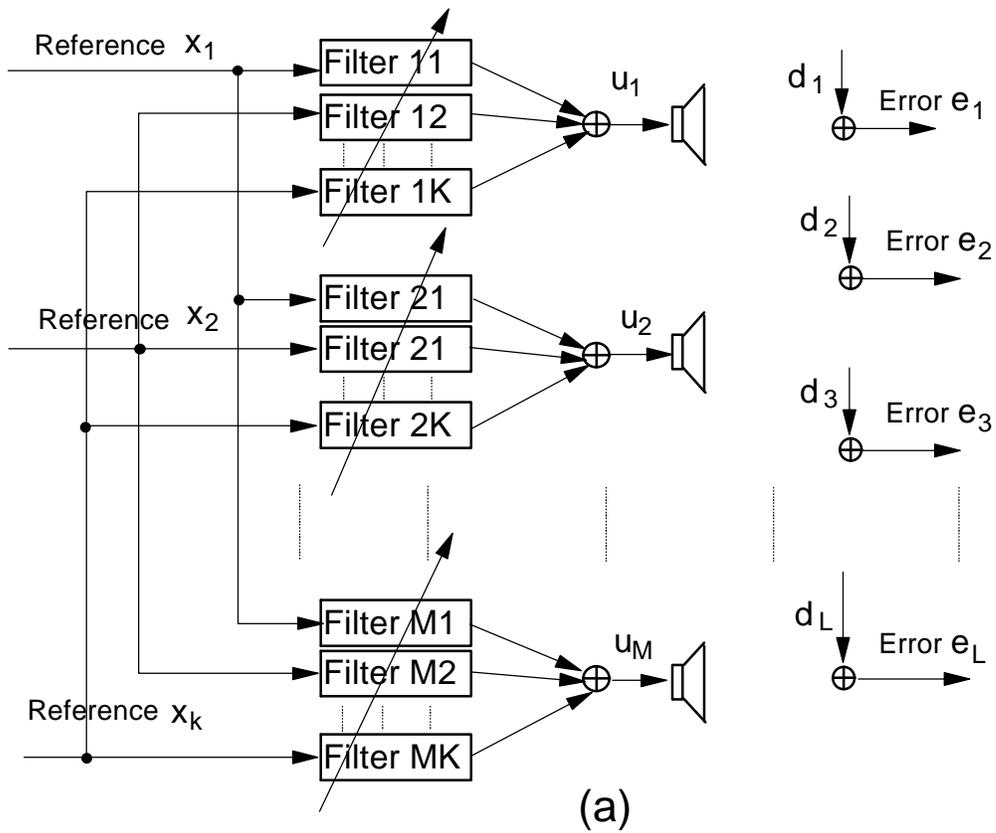
Figure 3.9 Multiple reference multiple channel active noise control system.

41

3.2 Multiple Channel Control System

Many active noise control applications involve the attenuation of low frequency noise in a simple spatial field. This allows the application of a single channel ANC system to achieve desirable noise reduction. In a large dimension enclosure where acoustical field is complicated, it is generally not enough to adjust a single secondary source to minimize a single error signal because noise cancellation at one point is likely to lead to the noise enhancement at other points [8]. In principle, the number of secondary sources needed to obtain perfect noise cancellation in an enclosure is the same as the number of acoustic modes excited [10]. Therefore, a multiple channel ANC system is needed to achieve global noise cancellation in an enclosure with a complex noise field.

3.2.1 Frequency Domain Optimum Solution

A multiple reference multiple channel control system is illustrated in Figure 3.9. The analysis procedure presented in this section follows closely that presented in section 3.1.1.

The signal at the lth error sensor is contributed from both primary noise sources and secondary control sources and can be written as

$$e_l(w) = d_l(w) + \sum_{m=1}^{M} T_{ml}(w)u_m(w) \tag{3.2.1}$$

where $T_{ml}(w)$ is the transfer function between the mth secondary source and the lth error sensor, $d_l(w)$ is the signal at the lth error sensor due to the primary sources. The signal $u_m(w)$ is the summation of the filter output signals to the mth secondary source, which is given by

$$u_m(w) = \sum_{k=1}^{K} x_k(w)W_{km}(w) \tag{3.2.2}$$

where K is the number of the reference signals. Substituting the above equation into equation (3.2.1) yields

$$e_l(w) = d_l(w) + \sum_{m=1}^{M} T_{ml}(w) \sum_{k=1}^{K} x_k(w)W_{km}(w)$$

$$= d_l(w) + \sum_{m=1}^{M} \sum_{k=1}^{K} T_{ml}(w)x_k(w)W_{km}(w) \tag{3.2.3}$$

Again, defining the filtered reference signal as

$$\hat{x}_{klm}(w) = x_k(w)T_{ml}(w)$$

equation (3.2.3) simplifies to

$$e_l(w) = d_l(w) + \sum_{i=1}^{K} \hat{x}_{klm}(w)W_{km}(w)$$

$$= d_l(w) + \hat{\mathbf{x}}_l^{T}(w)\mathbf{W}(w) \tag{3.2.4}$$

where

$$\hat{\mathbf{x}}_l(w) = \left\{ \hat{x}_{11l} \quad \hat{x}_{12l} \quad \cdots \quad \hat{x}_{1ml} \quad | \quad \hat{x}_{21l} \quad \hat{x}_{22l} \quad \cdots \quad \hat{x}_{2ml} \right.$$

$$\{ \quad | \quad \cdots \quad | \quad \hat{x}_{k1l} \quad \hat{x}_{k2l} \quad \cdots \quad \hat{x}_{kml} \} \tag{3.2.5}$$

$$\mathbf{W}(w) = \{ W_{11} \quad W_{12} \quad \cdots \quad W_{1m} \quad | \quad W_{21} \quad W_{22} \quad \cdots \quad W_{2m}$$

$$\qquad\qquad | \quad \cdots \quad | \quad W_{k1} \quad W_{k2} \quad \cdots \quad W_{km} \} \tag{3.2.6}$$

Next, the error vector is defined as

$$\mathbf{e}(w) = \{ e_1(w) \quad e_2(w) \quad \cdots \quad e_L(w) \}^H \tag{3.2.7}$$

Using equation (3.2.4), error vector can be expressed as

$$\mathbf{e}(w) = \mathbf{d}(w) + \hat{\mathbf{x}}(w)\mathbf{W}(w) \tag{3.2.8}$$

where

$$\mathbf{d}(w) = \{ d_1(w) \quad d_2(w) \quad \cdots \quad d_L(w) \}^T \tag{3.2.9}$$

$$\hat{\mathbf{x}}(w) = \{ \hat{\mathbf{x}}^T{}_1(w) \quad \hat{\mathbf{x}}^T{}_2(w) \quad \cdots \quad \hat{\mathbf{x}}^T{}_K(w) \}^T \tag{3.2.10}$$

The active noise control objective is to minimize overall sound pressure level, which is estimated by the mean square error signals. Thus the cost function can be constructed as the summation of the power spectrum of all the error signals, that is

$$\xi(w) = \mathrm{E}\big(\mathbf{e}^H(w)\mathbf{e}(w) \big) \tag{3.2.11}$$

Substituting equation (3.2.8) into the above equation yields

$$\xi(w) = \mathrm{E}\big(\mathbf{d}^H(w)\mathbf{d}(w) \big) + \mathrm{E}\big(\mathbf{d}^H(w)\mathbf{x}(w)\mathbf{W}(w) \big) +$$

$$\mathrm{E}\big(\mathbf{W}^H(w)\mathbf{x}^H(w)\mathbf{d}(w) \big) + \mathrm{E}\big(\mathbf{W}^H(w)\mathbf{x}^H(w)\mathbf{x}(w)\mathbf{W}(w) \big) \tag{3.2.12}$$

The above cost function has *Hermitian quadratic form* and is a real scalar. The gradient of the cost function with respect to the complex weight vector is expressed as

$$\nabla = \frac{\P\mathsf{x}(w)}{\P\mathbf{W}(w)}$$

$$= \mathrm{E}\big(\hat{\mathbf{x}}^H(w)\hat{\mathbf{x}}(w) \big)\mathbf{W}(w) + \mathrm{E}\big(\mathbf{d}(w)\hat{\mathbf{x}}^H(w) \big)$$

$$= \mathbf{R}(w)\mathbf{W}(w) + \mathbf{P}(w) \tag{3.2.13}$$

where

$$\mathbf{R}(w) = \mathbf{S}_{\mathbf{xx}}(w) = \mathrm{E}\big(\hat{\mathbf{x}}^H(w)\hat{\mathbf{x}}(w) \big) \tag{3.2.14}$$

$$\mathbf{P}(w) = \mathbf{S}_{\mathbf{xd}}(w) = \mathrm{E}\big(\mathbf{d}(w)\hat{\mathbf{x}}^H(w) \big) \tag{3.2.15}$$

The correlation matrix $\mathbf{R}(w)$ is generally positive definite, and the cost function $\xi(w)$ has a unique global minimum value. The weight vector corresponding to the minimum value of the cost function is referred to as the optimum weight vector, which is obtained by setting the complex gradient vector to zero. Thus,

$$\mathbf{W}_{opt}(w) = -\mathbf{R}^{-1}(w)\mathbf{P}(w) \tag{3.2.16}$$

3.2.2 Coherence Analysis

Equation (3.2.12) can be written as

$$S_{ee}(w) = S_{dd}(w) + \mathbf{S}_{d\hat{x}}(w)\mathbf{W}(w) + \mathbf{S}_{\hat{x}d}(w)\mathbf{W}^*(w) + \mathbf{W}^H(w)\mathbf{S}_{\hat{x}\hat{x}}(w)\mathbf{W}(w) \qquad (3.2.17)$$

when the controller is optimized with equation (3.2.16), the auto-spectrum of the error signal reaches its minimum value, which is obtained by substituting equation (3.2.16) into (3.2.17) as

$$S_{ee}(w) = S_{dd}(w) + \mathbf{S}^T_{d\hat{x}}(w)\mathbf{S}_{\hat{x}\hat{x}}^{-1}(w)\mathbf{S}_{\hat{x}d}(w) \qquad (3.2.18)$$

dividing both side of the equation (3.1.14) by $S_{dd}(w)$, the relative noise reduction is obtained as

$$\frac{S_{ee}(w)}{S_{dd}(w)} = 1 - \frac{\mathbf{S}^T_{d\hat{x}}(w)\mathbf{S}_{\hat{x}\hat{x}}^{-1}(w)\mathbf{S}_{\hat{x}d}(w)}{S_{dd}(w)}$$

$$= 1 - g^2_{\hat{x}d}(w) \qquad (3.2.19)$$

where $g^2_{\hat{x}d}(w)$ is the coherence function based on a multiple reference multiple error ANC system. Equation (3.2.19) indicates that the overall noise reduction is bounded It should be noted that the multiple coherence function defined in equation (3.1.22) is based on a multiple reference single error ANC system.

3.2.3 Optimum FIR Filter

As is pointed out in the first chapter, a frequency domain optimum filter is not guaranteed to be causal, therefore it may not be realizable in a real system. Usually, the controller is implemented using a FIR filter. The corresponding optimum solution can be derived as follows:

At time step k, the signal at the lth error sensor can be written as

$$e_l(k) = d_l(k) + \sum_{m=1}^{M} T_{ml}(z)u_m(k) \qquad (3.2.20)$$

where $T_{ml}(z)$ is the transfer function between the mth secondary source and the lth error sensor in the Z domain. The signal $u_m(k)$ is the summation of the filter output signals to the mth secondary source, given by

$$u_m(k) = \sum_{k=1}^{K} \mathbf{x}^T_k(k)\mathbf{w}_{km} \qquad (3.2.21)$$

where K is the number of the reference signals, $\mathbf{x}_k(k)$ is the input vector at the reference k and $\mathbf{w}_{km}$ is the weight vector from the reference k to the secondary source m. Assuming that the vector length is I, $\mathbf{x}_k(k)$ and $\mathbf{w}_{km}$ become

$$\mathbf{x}_k(k) = \{x_k(k) \quad x_k(k-1) \quad x_k(k-2) \quad \cdots \quad x_k(k-I+1)\}^T \qquad (3.2.22)$$

$$\mathbf{w}_{km} = \{w_{km}(0) \quad w_{km}(1) \quad w_{km}(2) \quad \cdots \quad w_{km}(I-1)\}^T \qquad (3.2.23)$$

Substituting equation (3.2.21) into (3.2.20) yields

$$e_l(k) = d_l(k) + \sum_{m=1}^{M} T_{ml}(z)\sum_{k=1}^{K} \mathbf{x}_k(k)\mathbf{w}_{km}$$

$$= d_l(k) + \sum_{m=1}^{M} \sum_{k=1}^{K} T_{ml}(z) \mathbf{x}^T_k(k) \mathbf{w}_{km} \qquad (3.2.24)$$

Again, defining the filtered reference signal as

$$\hat{\mathbf{x}}_{klm}(k) = \mathbf{x}_k(k) T_{ml}(z) \qquad (3.2.25)$$

equation (3.2.24) simplifies to

$$e_l(k) = d_l(k) + \sum_{m=1}^{M} \sum_{k=1}^{K} \hat{\mathbf{x}}^T_{kml}(k) \mathbf{w}_{km}$$

$$= d_l(k) + \hat{\mathbf{X}}_l^T(k) \mathbf{W} \qquad (3.2.26)$$

where

$$\hat{\mathbf{X}}_l(k) = \left\{ \hat{\mathbf{x}}^T_{11l} \quad \hat{\mathbf{x}}^T_{12l} \quad \cdots \quad \hat{\mathbf{x}}^T_{1ml} \quad | \quad \hat{\mathbf{x}}^T_{21l} \quad \hat{\mathbf{x}}^T_{22l} \quad \cdots \quad \hat{\mathbf{x}}^T_{2ml} \right.$$

$$\left. | \quad \cdots \quad | \quad \hat{\mathbf{x}}^T_{k1l} \quad \hat{\mathbf{x}}^T_{k2l} \quad \cdots \quad \hat{\mathbf{x}}^T_{kml} \right\}^T \qquad (3.2.27)$$

$$\mathbf{W} = \left\{ \mathbf{w}^T_{11} \quad \mathbf{w}^T_{12} \quad \cdots \quad \mathbf{w}^T_{1m} \quad | \quad \mathbf{w}^T_{21} \quad \mathbf{w}^T_{22} \quad \cdots \quad \mathbf{w}^T_{2m} \right.$$

$$\left. | \quad \cdots \quad | \quad \mathbf{w}^T_{k1} \quad \mathbf{w}^T_{k2} \quad \cdots \quad \mathbf{w}^T_{km} \right\}^T \qquad (3.2.28)$$

Next, the error vector is defined as

$$\mathbf{e}(k) = \left\{ e_1(k) \quad e_2(k) \quad \cdots \quad e_L(k) \right\}^T \qquad (3.2.29)$$

Using equation (3.2.4), error vector can be expressed as

$$\mathbf{e}(k) = \mathbf{d}(k) + \hat{\mathbf{X}}(k) \mathbf{W} \qquad (3.2.30)$$

where

$$\mathbf{d}(k) = \left\{ d_1(k) \quad d_2(k) \quad \cdots \quad d_L(k) \right\}^T \qquad (3.2.31)$$

$$\hat{\mathbf{X}}(k) = \left\{ \hat{\mathbf{X}}^T_1(k) \quad \hat{\mathbf{X}}^T_2(k) \quad \cdots \quad \hat{\mathbf{X}}^T_L(k) \right\}^T \qquad (3.2.32)$$

The goal of the active noise control system is to minimize the total acoustical potential energy, which is estimated by the mean square error signals. Thus the cost function is constructed as

$$x = E\left(\mathbf{e}^T(k) \mathbf{e}(k) \right) \qquad (3.2.33)$$

Substituting (3.2.30) into the above equation yields

$$x = E\left(\mathbf{d}^T(k) \mathbf{d}(k) \right) + E\left(\mathbf{d}^T(k) \hat{\mathbf{X}}(k) \mathbf{W} \right) +$$

$$E\left(\mathbf{W}^T(k) \mathbf{X}^T(k) \mathbf{d}(k) \right) + E\left(\mathbf{W}^T(k) \mathbf{X}^T(k) \mathbf{X}(k) \mathbf{W}(k) \right) \qquad (3.2.34)$$

Defining the matrix $\mathbf{R}$ and the vector $\mathbf{P}$ as

$$\mathbf{R} = E\left(\hat{\mathbf{X}}^T(k) \hat{\mathbf{X}}(k) \right) \qquad (3.2.35)$$

$$\mathbf{P} = E\left(\hat{\mathbf{X}}^T(k) \mathbf{d}(k) \right) \qquad (3.2.36)$$

equation (3.2.34) can be rewritten as

$$x = E\left(\mathbf{d}^T(k) \mathbf{d}(k) \right) + 2\mathbf{P}^T \mathbf{W} + \mathbf{W}^T \mathbf{R} \mathbf{W} \qquad (3.2.37)$$

Again, it is noted that the above cost function has *Hermitian quadratic form*, and is a real scalar. The gradient of the cost function with respect to the weight vector can be obtained as

$$\nabla = \mathbf{R} \mathbf{W} + \mathbf{P} \qquad (3.2.38)$$

45

If the correlation matrix $\mathbf{R}$ has an inverse, which indicates a unique global minimum for the cost function, the optimum weight vector can be obtained by setting the gradient to zero as

$$\mathbf{W}_{opt} = -\mathbf{R}^{-1}\mathbf{P} \tag{3.2.39}$$

Substituting equation (3.2.38) into (3.2.24), the corresponding minimum mean square error is expressed as

$$\chi_{\min} = E\left(\mathbf{d}^T(k)\mathbf{d}(k)\right) - \mathbf{P}^T\mathbf{W}_{opt} \tag{3.2.40}$$

and equation (3.2.37) can be expressed as

$$\chi = \chi_{\min} + \left(\mathbf{W} - \mathbf{W}_{opt}\right)^T \mathbf{R}\left(\mathbf{W} - \mathbf{W}_{opt}\right) \tag{3.2.41}$$

The above two equations will be used in the next chapter to derive the misadjustment for the DFXLMS algorithm.

3.2.4 On the Problem of More Secondary Sources Than Error Sensors

It has been pointed out that the choice of an ANC system with more secondary sources than error sensors should be avoided because it leads to the problem of an underdetermined system [35]. An underdetermined system has infinite number of optimum solutions for the weight vector. Some of the solutions are associated with very large control effort, which is generally not desired.

A simple system with 1 reference, 2 secondary sources and 1 error sensor is considered as shown in Fig. 3.10a. As mentioned earlier, such a system is commutable between the adaptive filter and the error path provided that the adaptive filter changes slowly compared with the response time of the error path. The commuted system, shown in Fig 3.10b, exhibits strong similarity with the multiple reference single channel system, shown in Figure 3.3.

The correlation matrix of a multiple reference single channel system has been obtained in equation (3.1.43). Since there are only two filtered reference signals in Fig. 3.10b, equation (3.1.43) can be simplified as

$$\mathbf{R} = E\begin{bmatrix} \hat{X}_1(k)\hat{X}_1(k) & \hat{X}_1(k)\hat{X}_2(k) \\ \hat{X}_2(k)\hat{X}_1(k) & \hat{X}_2(k)\hat{X}_2(k) \end{bmatrix} \tag{3.2.42}$$

It should be noted that every term inside the above matrix is a sub-matrix. the influence of the individual sub-matrix to the property of the correlation matrix $\mathbf{R}$ is ignored and the attention is focused on the cross-correlation of these matrices.

The filtered reference signals $\hat{x}_1$ and $\hat{x}_2$ are derived from the same reference signal x_1 through different error paths, therefore $\hat{x}_1$ and $\hat{x}_2$ are correlated more or less depending on the characteristics of the error paths. If the transfer functions of two error paths have no overlapped frequency band (e.g. one is high-pass, the other is low pass), $\hat{x}_1$ and $\hat{x}_2$ will be uncorrelated. In this case, the system behaves like multiple reference ANC system with two independent

references. However, such a case is scarce, usually $\hat{x}_1$ and $\hat{x}_2$ are strongly correlated due to the close location of error paths. This correlation causes the two columns in the above matrix R similar and leads to the problem of severe ill-conditioning.

If the difference between the two error paths is ignored, which implies that the two filtered reference signals $\hat{x}_1$ and $\hat{x}_2$ are exactly the same and the two columns in the above $\boldsymbol{R}$ matrix are linearly dependent, the system becomes underdetermined. It should be noted that underdetermining is the extreme case of ill-conditioning. As a conclusion, the ANC system with more secondary sources than error sensors is an ill-conditioned system, which usually requires large convergence time, especially when the FXLMS algorithm is applied.
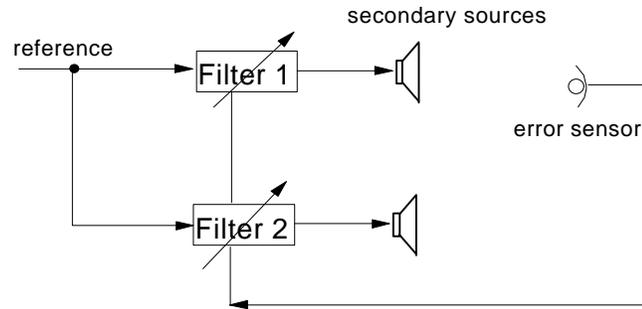
(a)
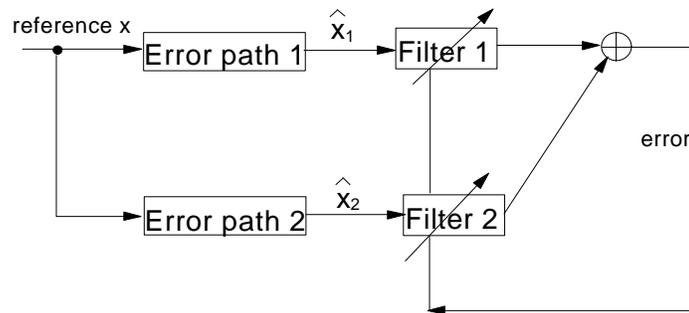
(b)

Figure 3.10 An ANC system with more secondary sources than error sensors, (a) standard diagram, (b) diagram with commutation between error path and adaptive filter.

4 Implementation of the DFXLMS Algorithm

This chapter discusses the real time implementation of a multiple reference multiple channel active noise control system. First, the multiple reference *Decorrelated Filtered-X LMS* (DFXLMS) algorithm is developed. This algorithm differs from the traditional FXLMS algorithm in that it decorrelates all the reference signals by preprocessing them through a set of adaptive filters. Then, the convergence behavior and the misadjustment due to the gradient noise of the DFXLMS algorithm are studied and compared with the traditional FXLMS algorithm. Furthermore, several features of the special purpose microprocessor TMS320C30 are introduced, and the discussions on how to efficiently take advantage of these features for the DFXLMS algorithm are also presented. Finally, because there are a large number of real time data to be processed in the algorithm and large number of adjustable parameters for the system, a graphical user interface (GUI) is provided to facilitate data monitoring, parameter modification as well as DSP process control.

4.1 The Multiple Reference DFXLMS Algorithm

4.1.1 The Multiple Reference DFXLMS Algorithm

A block diagram of the DFXLMS algorithm is shown in Fig. 4.1, in which there are K reference signals, M secondary sources, L error sensors. The number of the adaptive filters and the number of error paths are K×M and M×L respectively. In this configuration, each secondary source is capable of canceling all the noises detectable through K reference sensors. It is obvious from the structure that the algorithm can be split into two sections. In the first section, by pre-filtering the reference signal x_1, x_2, $\cdots$, x_K, a set of mutually uncorrelated reference signals $\bar{x}_1$, $\bar{x}_2$, $\cdots$, $\bar{x}_K$ are obtained, and the second section deals with the traditional Filtered-X LMS algorithm.

The first section is composed of $k \times (k+1)$ adaptive transversal FIR filters as shown in Figure 3.5. The first output signal is obtained as

$$\bar{x}_1(k) = x_1(k) + \bar{\mathbf{x}}_2(k) \otimes \mathbf{B}_{21}(k) + \cdots + \bar{\mathbf{x}}_k(k) \otimes \mathbf{B}_{k1}(k) \qquad (4.1.1)$$

where the symbol $\otimes$ denotes convolution. Generally, the ith output signal can be obtained as

$$\bar{x}_i(k) = x_i(k) + \bar{\mathbf{x}}_1(k) \otimes \mathbf{A}_{i1}(k) + \cdots + \bar{\mathbf{x}}_{i-1}(k) \otimes \mathbf{A}_{i(i-1)}(k) +$$
$$\bar{\mathbf{x}}_{i+1}(k) \otimes \mathbf{B}_{(i+1)i}(k) + \cdots + \bar{\mathbf{x}}_k(k) \otimes \mathbf{B}_{ki}(k) \qquad (4.1.2)$$

The last output signal is obtained

$$\bar{x}_k(k) = x_k(k) + \bar{\mathbf{x}}_1(k) \otimes \mathbf{A}_{k1}(k) + \cdots + \bar{\mathbf{x}}_{k-1}(k) \otimes \mathbf{A}_{k(k-1)}(k) \qquad (4.1.3)$$

As discussed in the Section 3.1.4, **A**s and **B**s vectors have redundancy in their first coefficients. To avoid this redundancy, the first coefficients in either **A**s or **B**s vectors should be set to zero.
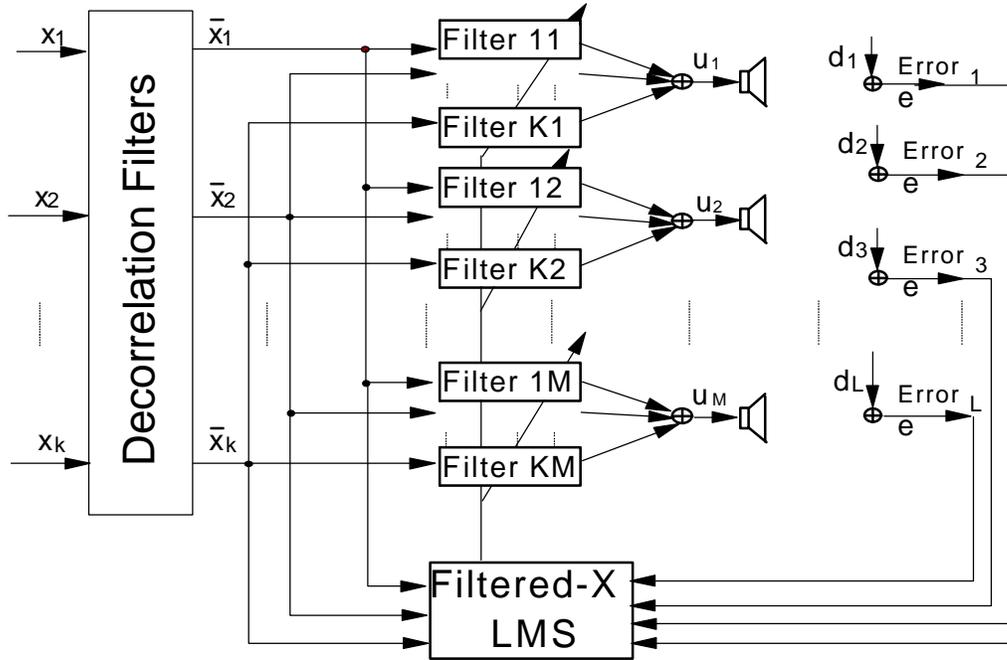
Figure 4.1 Block diagram of multiple reference decorrelated Filtered-X LMS algorithm.

Assuming that the coefficients in **B**s vectors are set to zero, and if the standard LMS algorithm is applied to the decorrelation filters, the weight vectors can be updated as

$$b_{mn}(k+1,i) = b_{mn}(k,i) - \mu \bar{x}_m(k-i)\bar{x}_n(k) \qquad i = 1, 2, \cdots M-1 \qquad (4.1.4)$$

$$a_{mn}(k+1,i) = a_{mn}(k,i) - \mu \bar{x}_m(k-i)\bar{x}_n(k) \qquad i = 0, 1, 2, \cdots M-1 \qquad (4.1.5)$$

where M is the number of coefficients for each decorrelation FIR filter. Now the mutually uncorrelated signals $\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_K$ are obtained in the above equations. The next step is to derive the traditional multiple reference Filtered-X LMS algorithm. The error signal at the lth error sensor is obtained in equation (3.2.26), which can be rewritten as

$$e_l(k) = d_l(k) + \sum_{m=1}^{M}\sum_{k=1}^{K} \hat{\mathbf{x}}^T_{kml}(k)\mathbf{w}_{km} \qquad\qquad l = 1,2,\cdots,L$$

$$= d_l(k) + \hat{\mathbf{X}}_l^{T}(k)\mathbf{W} \qquad\qquad (4.1.6)$$

where the symbol $\wedge$ denotes the filtered reference signal, and the subscript *ml* denotes the error path from the mth secondary source to the lth error sensor. The cost function is constructed as

$$\chi(k) = E\left(\sum_{l=1}^{L}\left(e^2_l(k)\right) \right) \qquad\qquad (4.1.7)$$

The gradient of the cost function with respect to the whole weight vector **W** is estimated using the instantaneous value and is given by

$$\nabla(k) = \frac{\partial \chi(k)}{\partial \mathbf{W}} \cong \sum_{l=1}^{L} 2e_l(k)\hat{\mathbf{X}}_l(k) \qquad\qquad (4.1.8)$$

The weight vector updating equation is obtained as

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu\sum_{l=1}^{L}\left(2e_l(k)\hat{\mathbf{X}}_l(k)\right) \qquad\qquad (4.1.9)$$

where

$$\mathbf{X}_l(k) = \sum_{m=1}^{M}\sum_{k=1}^{K}\bar{\mathbf{x}}_k(k)T_{ml}(z)$$

and the output signal to the mth secondary source is the summation of the output signals from K filters, which can be written as

$$u_m(k) = \sum_{k=1}^{K}\bar{\mathbf{x}}^T_k(k)\mathbf{w}_{km}(k) \qquad m = 1,2,\cdots,M \qquad (4.1.10)$$

Equations (4.1.9) and (4.1.10) form the FXLMS algorithm for the multiple reference multiple channel ANC system shown in Fig. 4.1.

4.1.2 Convergence Analysis of the Multiple Reference DFXLMS Algorithm

It is well-known that the convergence speed of the LMS algorithm is inversely proportional to the eigenvalue spread [23]. The eigenvalue spread is determined by both the auto-correlation and the cross-correlation of the reference signals. If the eigenvalue spread is large, the convergence speed becomes slow. The DFXLMS weight updating is obtained in equation (4.1.9), which can be rewritten as

$$\mathbf{W}(k+1) = \mathbf{W}(k) - m \sum_{l=1}^{L} \left(2 \left(d_l(k) + \hat{\mathbf{X}}^T(k) \mathbf{W} \right) \hat{\mathbf{X}}_l^T(k) \right) \tag{4.1.11}$$

Taking the expected value of both sides of the equation (2.3.9) and assuming that $\mathbf{W}$(k) and $\mathbf{X}$(k) are independent yield

$$E(\mathbf{W}(k+1)) = E(\mathbf{W}(k)) - m \left(\mathbf{P} + \mathbf{R} E(\mathbf{W}(k)) \right)$$

$$= E(\mathbf{W}(k)) - m \left(- \mathbf{R} \mathbf{W}_{opt} + \mathbf{R} E(\mathbf{W}(k)) \right) \tag{4.1.12}$$

where $\mathbf{W}_{opt}$ denotes the optimum weight vector defined in equation (3.2.38), $\mathbf{R}$ and $\mathbf{P}$ are the correlation matrix and the cross-correlation vector defined in equations (3.2.36) and (3.2.37) respectively. Defining the shifted weight vector as

$$\hat{\mathbf{W}}(k) = \mathbf{W}(k) - \mathbf{W}_{opt} \tag{4.1.13}$$

then

$$E(\mathbf{W}(k+1)) = (\mathbf{I} - m\mathbf{R}) E(\mathbf{W}(k)) \tag{4.1.14}$$

The auto-correlation matrix $\mathbf{R}$ is symmetrical, thus it can be transformed into a diagonal matrix through a unitary transformation [36].

$$\mathbf{R} = \mathbf{Q} \Lambda \mathbf{Q}^T \tag{4.1.15}$$

where Λ is a diagonal matrix, and its diagonal components are the eigenvalues of the matrix $\mathbf{R}$. $\mathbf{Q}$ is a unitary matrix, whose columns are the normalized eigenvectors of the matrix $\mathbf{R}$. Substituting equation (2.3.13) into equation (2.3.12) and using $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}$ yield

$$E(\hat{\mathbf{W}}(k+1)) = \mathbf{Q}(\mathbf{I} - m\Lambda)\mathbf{Q}^T E(\hat{\mathbf{W}}(k)) \tag{4.1.16}$$

Defining the primed weight vector as

$$\mathbf{V}(k) = \mathbf{Q}^T \hat{\mathbf{W}}(k) \tag{4.1.17}$$

equation (2.3.14) becomes

$$E(\mathbf{V}(k+1)) = (\mathbf{I} - m\Lambda) E(\mathbf{V}(k)) \tag{4.1.18}$$

Since the matrix $\mathbf{R}$ is non-negative definite, all its eigenvalues are non-negative. In order for the primed weight vector $\mathbf{V}$ to converge, the convergence parameter μ has to be bounded such that

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{4.1.19}$$

If the eigenvalue spread ($\lambda_{max}/\lambda_{min}$) is large, although it could takes only one step for the primed weight corresponding to the maximum eigenvalue to converge, it takes many steps to converge for the weight corresponding to the minimum eigenvalue. Therefore, the overall convergence rate for the weight vector is still slow. Since the DFXLMS algorithm uses uncorrelated reference signals, its eigenvalue spread are expected to be smaller compared to the conventional FXLMS algorithm. It follows that the convergence rate of the DFXLMS algorithm is faster than the FXLMS algorithm.

4.1.3 Gradient Noise and Misadjustment of the multiple reference DFXLMS Algorithm

If the convergence condition is satisfied, the weight vector will converge to the optimum Wiener solution. However, the mean square error (MSE) will not converge to the minimum. In

fact, the MSE of the DFXLMS algorithm is always larger than the minimum, as is the case for the LMS algorithm. This is due to the gradient noise in the weight updating equation, which is rewritten as

$$\mathbf{W}(k+1) = \mathbf{W}(k) - m(\nabla - \mathbf{N}(k)) \qquad (4.1.20)$$

where $\mathbf{N}$ is the gradient noise. When the weight vector converges, the true gradient goes to zero, and the gradient estimation used in the weight updating equation is equal to the gradient noise, i.e.

$$\mathbf{N}(k) = \sum_{l=1}^{L} \left(2e_l(k)\hat{\mathbf{X}}_l(k) \right) \qquad (4.1.21)$$

and ∇ is the true gradient and can be written as

$$\nabla(k) = 2\left(\mathbf{P} + \mathbf{R}\mathbf{W}(k)\right)$$
$$= 2\mathbf{R}\left(\mathbf{W}(k) - \mathbf{W}_{opt}\right) \qquad (4.1.22)$$

Substituting equations (4.1.21) and (4.1.22) into (4.1.20) yields

$$\mathbf{W}(k+1) = \mathbf{W}(k) - m\left(2\mathbf{R}\left(\mathbf{W}(k) - \mathbf{W}_{opt}\right) - \mathbf{N}(k)\right) \qquad (4.1.23)$$

Transforming into the primed coordinates using equations (4.1.13) and (4.1.17), equation (4.1.23) can be rewritten as

$$\mathbf{V}(k+1) = \mathbf{V}(k) - m(2\Lambda\mathbf{V}(k) - \tilde{\mathbf{N}}(k))$$
$$= \left(1 - 2m\Lambda\right)\mathbf{V}(k) - \tilde{\mathbf{N}}(k) \qquad (4.1.24)$$

where $\tilde{\mathbf{N}}(k)$ is the gradient noise in the primed coordinates. According to the Wiener filter theory, when the weight vector converges to the optimum Wiener filter solution, $\hat{\mathbf{X}}_l(k)$ and $e_l(n)$ are uncorrelated. If assuming that the disturbance signals are Gaussian noise, then $\hat{\mathbf{X}}_l(k)$ and $e_l(n)$ are independent. Furthermore, two additional assumptions are made here that, upon convergence, 1) error signals at each error sensor have the same amplitude, 2) the filtered reference signals have the same amplitude. Then, the covariance matrix of $\mathbf{N}$(k) is obtained as

$$\mathrm{cov}\left(\mathbf{N}(k)\right) = \mathrm{E}\left(\mathbf{N}(k)\mathbf{N}^T(k)\right)$$
$$= \mathrm{E}\left(\sum_{l=1}^{L} 2e_l(k)\hat{\mathbf{X}}_l(k) \sum_{l=1}^{L} 2e_l(k)\hat{\mathbf{X}}_l^T(k) \right)$$
$$= 4\sum_{l=1}^{L} \mathrm{E}\left(e^2{}_l(k)\right) \sum_{l=1}^{L} \mathrm{E}\left(\hat{\mathbf{X}}_l(k)\hat{\mathbf{X}}_l^T(k)\right)$$
$$= 4x_{\min}\mathbf{R} \qquad (4.1.25)$$

Projecting the gradient noise into the primed coordinates

$$\tilde{\mathbf{N}}(k) = \mathbf{Q}^{-1}\mathbf{N}(k) \qquad (4.1.26)$$

Then, the covariance of $\tilde{\mathbf{N}}(k)$ becomes

$$\mathrm{cov}\left(\tilde{\mathbf{N}}(k)\right) = x_{\min}\Lambda \qquad (4.1.27)$$

The covariance of $\mathbf{V}$(k) can be expressed as

$$cov(\mathbf{V}(k)) = E(\mathbf{V}(k)\mathbf{V}^T(k))$$
$$= E((\mathbf{I} - 2m\Lambda)\mathbf{V}(k)\mathbf{V}^T(k)(\mathbf{I} - 2m\Lambda)) + m^2 E(\tilde{\mathbf{N}}(k)\tilde{\mathbf{N}}^T(k)) +$$
$$m E(\tilde{\mathbf{N}}(k)\mathbf{V}^T(k)(\mathbf{I} - 2m\Lambda)) + m E((\mathbf{I} - 2m\Lambda)\mathbf{V}(k)\tilde{\mathbf{N}}^T(k))$$
$$= (\mathbf{I} - 2m\Lambda)E(\mathbf{V}(k)\mathbf{V}^T(k))(\mathbf{I} - 2m\Lambda) + m^2 E(\tilde{\mathbf{N}}(k)\tilde{\mathbf{N}}^T(k)) \qquad (4.1.28)$$

Since the covariance of $\mathbf{V}$(k+1) is equal to the covariance of $\mathbf{V}$(k), it is obtained that

$$cov(\mathbf{V}(k)) = (\mathbf{I} - 2m\Lambda)cov(\mathbf{V}(k))(\mathbf{I} - 2m\Lambda) + m^2 cov(\tilde{\mathbf{N}}(k)) \qquad (4.1.29)$$

Furthermore, since the components of $\mathbf{V}$(k) are mutually uncorrelated, and the covariance of $\mathbf{V}$(k) is diagonal, thus

$$cov(\mathbf{V}(k)) = (\mathbf{I} - 2m\Lambda)^2 cov(\mathbf{V}(k)) + m^2 4\chi_{min}\Lambda \qquad (4.1.30)$$

The above equation (4.1.30) can be rewritten

$$(\mathbf{I} - m\Lambda)cov(\mathbf{V}(k)) = m\chi_{min}\mathbf{I} \qquad (4.1.31)$$

Supposing that the convergence parameter is chosen to be small, i.e. $m\Lambda \ll \mathbf{I}$, the covariance of $\mathbf{V}$(k) is further simplified to

$$cov(\mathbf{V}(k)) = m\chi_{min}\mathbf{I} \qquad (4.1.32)$$

Since each weight noise is uncorrelated with other weight noises, the average MSE is equal to the reference signal power multiplied by the sum of the variances of a single weight noise. Accordingly,

$$(Ave.\,excess\;MSE) = E(x_i^2(k)) \cdot n \cdot k \cdot m \cdot \begin{pmatrix} \text{variance of single} \\ \text{weight noise} \end{pmatrix} \qquad (4.1.33)$$

where n is the filter length, k is the number of the reference signals, and m is the number of the secondary source. From equation (4.1.33), the variance of a single weight noise is $m\chi_{min}$, and the trace, i.e. the summation of all the diagonal terms, of the correlation matrix $\mathbf{R}$ is

$$tr(\mathbf{R}) = E(x^2{}_i(k)) \cdot n \cdot k \cdot m \qquad (4.1.34)$$

Thus, the excess MSE is obtained as

$$(Ave.\,excess\;MSE) = tr(\mathbf{R})m\chi_{min} \qquad (4.1.35)$$

and the misadjustment of the DFXLMS and FXLMS algorithm is

$$M = \frac{(Ave.\,excess\;MSE)}{\chi_{min}}$$
$$= tr(\mathbf{R})m \qquad (4.1.36)$$

Misadjustment tells how much the MSE after convergence deviates from the minimum MSE. The misadjustment is determined only by the convergence parameter μ and the trace of the correlation matrix $\mathbf{R}$, but not by the eigenvalue spread of the correlation matrix R. Thus, preprocessing the reference signals with the DFXLMS algorithm does not improve the misadjustment compared with the FXLMS algorithm. In addition, the minimum MSE does not depend on an adaptive algorithm, it is therefore expected that the FXLMS and DFXLMS algorithms have the same MSE after convergence.

4.2 Implementation with TMS320C30 DSP

It is noted that there are extensive amounts of computations associated with the DFXLMS algorithm. In real time control, the computations need to be carried out within each sample interval, the length of which is determined by the sampling frequency. Without a dedicated microprocessor, it is almost impossible to accomplish those computations while maintaining the required sampling rate.

Digital signal processors (DSP) are special purpose microprocessors with sophisticated bus structure and special instruction set. The TMS320C30 [37][38][39] is a high-performance CMOS 32-bit floating-point device with 60-ns single circle instruction execution time. The TMS320C30 can perform parallel multiplications and ALU operations on integer and floating point data in a single cycle, thus it has the functionality of parallel addressing. This feature could substantially improve computational speed since an adaptive algorithm is essentially comprised of a sequence of multiplications and additions.

Another significant feature is that the TMS320C30 provides a circular addressing mode. In the DFXLMS algorithm, extensive computations associated with convolution and correlation require the implementation of circular buffers in memory. A circular buffer is used to implement a sliding window that contains the most recent data to be processed. As new data is brought in, the new data overwrite the oldest data. This functionality can be achieved efficiently by a circular addressing mode without any explicit conditional operations.

It is noted that each adaptive filtering process can be divided into two steps: (1) filtering step, i.e. convoluting the tapped input vector with the filter weights to generate the filter output, (2) weight updating step, i.e. updating the weight vector according a specific adaptive algorithm. In the appendix A, two functions are introduced to carry out the functionality of each step. Based upon these two functions, implementation of the DFXLMS algorithm is more flexible and reliable.

It is also desirable for the DSP code to provide a random signal generator and a sine wave signal generator. These built-in signal generators can be used as noise sources to drive the primary speakers, in the mean time, the signals from the signal generators are also used as the reference signals in the ANC system. This enables broadband or single frequency experiments without external noise sources. Most importantly, the reference signals are completely coherent so that attention can be focused on other interested issues without worrying about the coherence. Moreover, since the primary noise is generated inside the DSP, the noise source is completely controllable. If some delays are added to the primary path through the DSP operation, the causality problem can be avoided.

TI optimizing C compiler, which takes advantage of the special features of the TMS320C30, is able to generate efficient assembly code. The frame of the DFXLMS algorithm is implemented using C language. The code written in C language generally has better maintainability and readability compared with that written in Assembly language, however, the

execution speed is compromised. To maximize the execution speed of the algorithm, the key parts of the algorithm are implemented using assembly language.

4.3 Graphical User Interface

There are many adjustable parameters in the DFXLMS algorithm, and the optimum values of those parameters vary with different applications. For some parameters, e.g. the convergence parameter μ and the sampling frequency f_s, it is almost impossible to assign the optimum values at the beginning. The process to obtain the optimum value is largely based on trial and error. There are also massive data needed to be processed in the DFXLMS algorithm. It is extremely helpful that these data be monitored, and measures be taken to control what is going on during the adaptive process.

Based on the above considerations, a graphical user interface running under Microsoft$^{\circledR}$ Windows is provided to fulfill those requirements. The GUI is also responsible for downloading, initializing and stopping the DSP code. As shown in Fig. 2.3, The main window of the GUI is composed of a title bar, a menu bar, a control bar, a status bar and several simplistic oscilloscopes.

The prominent feature of the main window is its control bar, which lies on the right side of the frame. The parameters in the control bar can be modified by one click or direct typing without pulling down the main menu and opening a dialog box. On the upper part of the control bar are three radio buttons, which represent the three status of the DSP. By default, the DSP status is idle, which means that there are only some I/O operations and no sophisticated computations involved. The estimation of error paths is performed if the DSP is running under the system identification status. The control status, by name, corresponds to the process when the DSP is trying to minimize the error signals. The other components in the control bar are for sampling frequency, control filters, decorrelation filters, display options and process controls.

In the main window are maximum eight simplistic oscilloscopes, which are used to display different data for different DSP status. When the DSP status is idle, the oscilloscopes simply display input and output data from and to I/O board. When the DSP status is system identification, the oscilloscopes display the error signals and the residual of the system identification. If the system identification is carried out correctly with sufficient number of filter coefficients, the residual of the system identification should be very small. Thus it can be seen whether the system identification is successful or not. When the DSP status is control, the oscilloscopes display error signals as well as the control output signals. Care should be given to watch these signal, if error signals go up, the DSP process should be immediately stopped, and the cause be examined carefully, otherwise, damage to the DSP board may occur. When there are more than 4 channels of inputs and outputs involved, the displayed channels can be switched by clicking the push buttons in the control bar. Thus, all the channels can be monitored.

The menu bar in the main window provides various options, for which there is no space to implement in the control bar. It is desirable to save various parameters after quitting the program so that the same parameters can be used next time. This is achieved with the file menu, which provides the functionality to open the default or other existing files and save files. The second menu is to download different DSP codes for different purposes. The third menu is dedicated to those parameters not included in the control bars. There are three sub-menus associated with the third menu, each of them bring up a dialog box. The dialog box brought up by configuration sub-menu enables the selection of the number of reference signals, the number of control channels and the number of error sensors. Different types of reference signals can also be selected. The dialog boxes brought up by system ID and control sub-menus is for selecting the FIR filter length, convergence parameters and et. al.. The fourth menu is to upload various signals from the DSP memory so that some post-process can be done for those data. The fifth menu is to display the coefficients of the control filter and the system identification filter in both the time and frequency domain. Looking at these coefficients graphically can give additional insight into the problem. For example, if most of the coefficients for the system identification are close to zero, the number of coefficients needed to identify the error path can be reduced. If anticipating the secondary path has a sharp resonance peak, while the frequency response of the FIR model is not sharp enough at resonance, more coefficients are probably needed. The last menu is to provide copyright information and on-line help. The status bar in the main window provides the running status information associated with the GUI as well as the DSP.

There are several error protection features with the GUI: (1) the system identification has to been done before real time control, and if any parameter associated with error paths identification (e.g. sampling frequency, the number of channels, or the number of errors) has been changed, the error paths have to be re-identified. (2) Any DSP has computational limits. Therefore, the computational sensitive parameters such as the sampling frequency, the filter length, the number of reference signals and/or the control channels cannot increase without bound. If the values of the parameters are too high such that the DSP computation limit is exceeded, the GUI will issue beeping signals. (3) When the analogue input voltage is out of range and causes the A/D converter get invalid data, the GUI issues a warning and exits immediately.
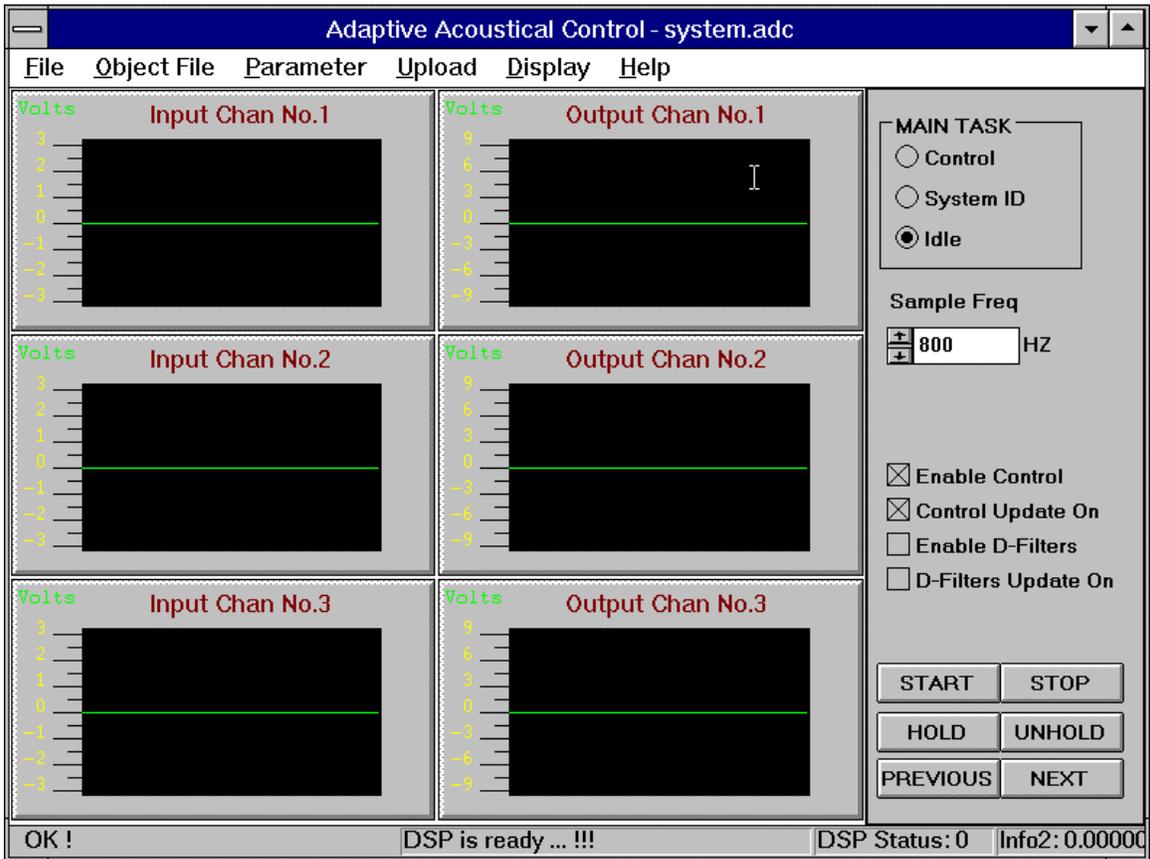
Figure 4.2 Main window of graphic user interface running under Microsoft® Windows.

Chapter 5 Simulation and Experimental Results

In the foregoing chapters, the algorithm (DFXLMS) and the configurations of multiple reference active noise control have been presented along with a discussion of various associated issues. In order to verify the algorithm and the configurations, extensive simulations and experiments were conducted on a vibrating plate with multiple primary excitations. The results corroborate the theoretical predictions: 1) the performance of the MRMI configuration is generally better than that of the MRSI configuration, 2) the DFXLMS algorithm can improve the condition number of the correlation matrix of the filtered reference signal, thus increasing the convergence speed compared to the conventional FXLMS algorithm.

In this chapter, first introduced is the test rig upon which both the simulations and the experiments are based. Then, the FIR models of the system transfer functions (primary path transfer functions and error path transfer function) are presented. Based upon the FIR models, computer simulations are conducted. Simulations are effective and efficient to investigate the algorithms and configurations. However, real systems are contaminated by the physical aspects such as measurement errors, non-coherent noise, conversion errors (A/D, D/A), non-linearity, non-stationarity and the like. Therefore, experimental implementation is very helpful to demonstrate the effectiveness and efficiency of the algorithm.

5.1 Test Rig

In a complicated ANC system, e.g. an aircraft cabin, the fuselage may generate noise due to direct applied forces as well as pressure variation. The test rig used here gives consideration to these two types of excitations. As shown in Figure 5.1, there are two disturbance sources, one secondary source and one error microphone for the plate system. The plate has dimensions of 0.381m long and 0.305m wide and is mounted in a heavy steel frame, which produces negligible rotation and displacement of the boundary, approximating clamped boundary conditions. The steel frame is mounted in a rigid wall with one side facing toward a reverberation chamber and the other side toward an anechoic chamber. The plate is excited by two distinctive noise sources; one is the acoustical disturbance from a large speaker, while the other is the structural disturbance from a piezoelectric ceramic/polymer composite transducer (PZT #1) mounted on the plate. The secondary control source acting on the plate is another PZT (PZT #2). The positions of the PZTs are selected such that any plate mode of order (4,4) or less can be excited {40] as shown in Figure 5.2. The error sensor is a microphone located in the direction approximately perpendicular to the center of the plate.
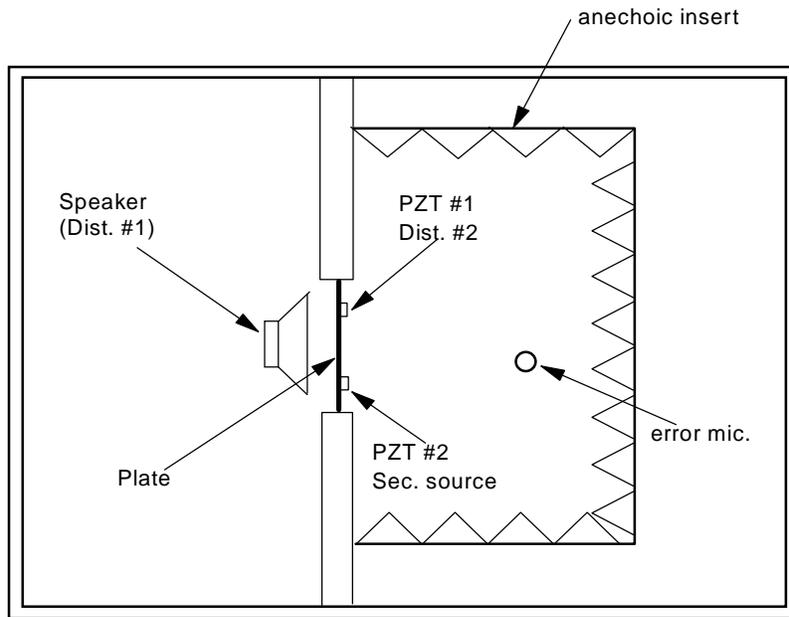
**Figure 5.1** System setup for the simulations and the experiments.
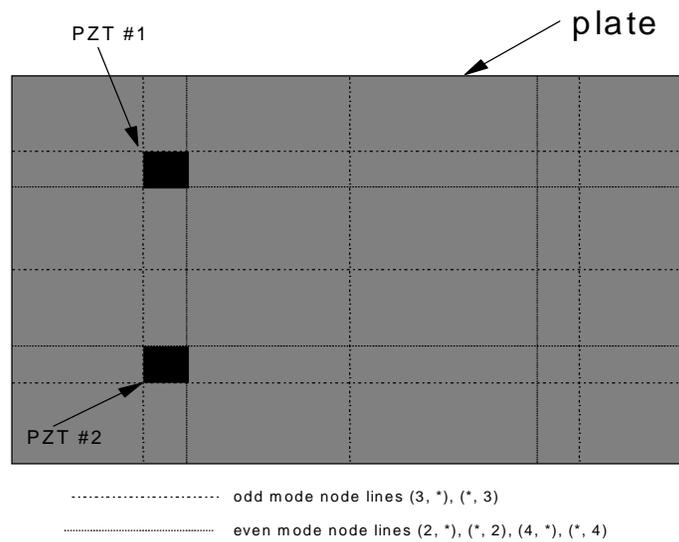
Figure 5.2 PZT locations and node lines for the tests.

5.2 Simulation Results

In this section, the theoretical analysis on a multiple reference ANC presented in the previous chapters is verified through various computer simulations. The first section discusses the modeling of the test rig using FIR filters. Then, based on the model, two control configurations discussed in section 3.1.5 are simulated. Finally, the performance of the DFXLMS algorithm is simulated and compared with the traditional FXLMS algorithm. All the simulations are carried out with MATLAB®, which provides built-in support for matrix operations and special DSP toolboxes to facilitate the implementation.

5.2.1 System Measurement and Modeling

The system shown in Figure 5.1 has two primary paths (from the speaker through the plate to the error microphone and from the PZT #1 to the error microphone) and one error path (from the PZT #2 to the error microphone). The transfer functions of the two primary paths and the error path were modeled with FIR filters. In order to obtain the FIR models of the system, the frequency response functions (FRF) haven been measured with a B&K 2032 digital signal analyzer, and 801 frequency response data points equally spaced between 0 to 400 Hz were obtained. These frequency response data were fitted with FIR filters using the least square method [41]. The frequency range is constrained to be below 400 Hz due to the limitation of the processing speed of the digital signal processor, since more FIR coefficients and higher sampling frequency are required to control a higher frequency range. The computational load is in direct proportional to the number of coefficients, therefore, when the frequency range increases, the DSP is required to perform more computations in a shorter period of time as determined by the sampling frequency. Within the chosen frequency range, a maximum of five structural modes can be excited. Each mode and its corresponding measured natural frequency are shown in Table 5.1.

Table 5.1 Excited modes and measured natural frequencies

Mode	(1.1)	(2.1)	(1.2)	(2.2)	(3.1)
Frequency (Hz)	115	201	265	342	350

There are 128 coefficients for each FIR filter. The number of coefficients is chosen such that both the phase and magnitude of the transfer function can be well matched at the frequency range where large noise cancellation is desired. The match at other frequency ranges (e.g. below 40 Hz) is not very important, since the noise cancellation at those frequency ranges is unobtainable due to the dynamic limitations of the system components (e.g. PZT and microphone). The sampling frequency is chosen to be 800 Hz which is exactly the Nyquist frequency for the system.

The primary transfer functions and the response of their corresponding FIR models are shown in Figure 5.3 and Figure 5.4. It should be noted that the two primary transfer functions are significantly different. The transfer function due to the acoustical disturbance from the speaker has dominant response at (1, 1) mode because the speaker induces an acoustical wave normal to the plate, while the transfer function due to the structural disturbance from the PZT has dominant response at modes (2,2) and (3,1) due to the location of the PZT #1 on the plate. The error path transfer function and its FIR model response are shown in Figure 5.5. The error path model is essential for the adaptive algorithms (FXLMS and DFXLMS) to estimate the correct direction of the cost function gradient. However, as long as the phase error between the error path and its model is within 90 degrees, the stability for both algorithms is assured [42].

It should be noted that the error path and the primary path due to the structural disturbance can be more efficiently (with less coefficients) modeled with IIR filters, especially at the resonance. However, since the experiments model the error path with a FIR filter, more comparable results can be obtained with the simulations based on FIR filters.
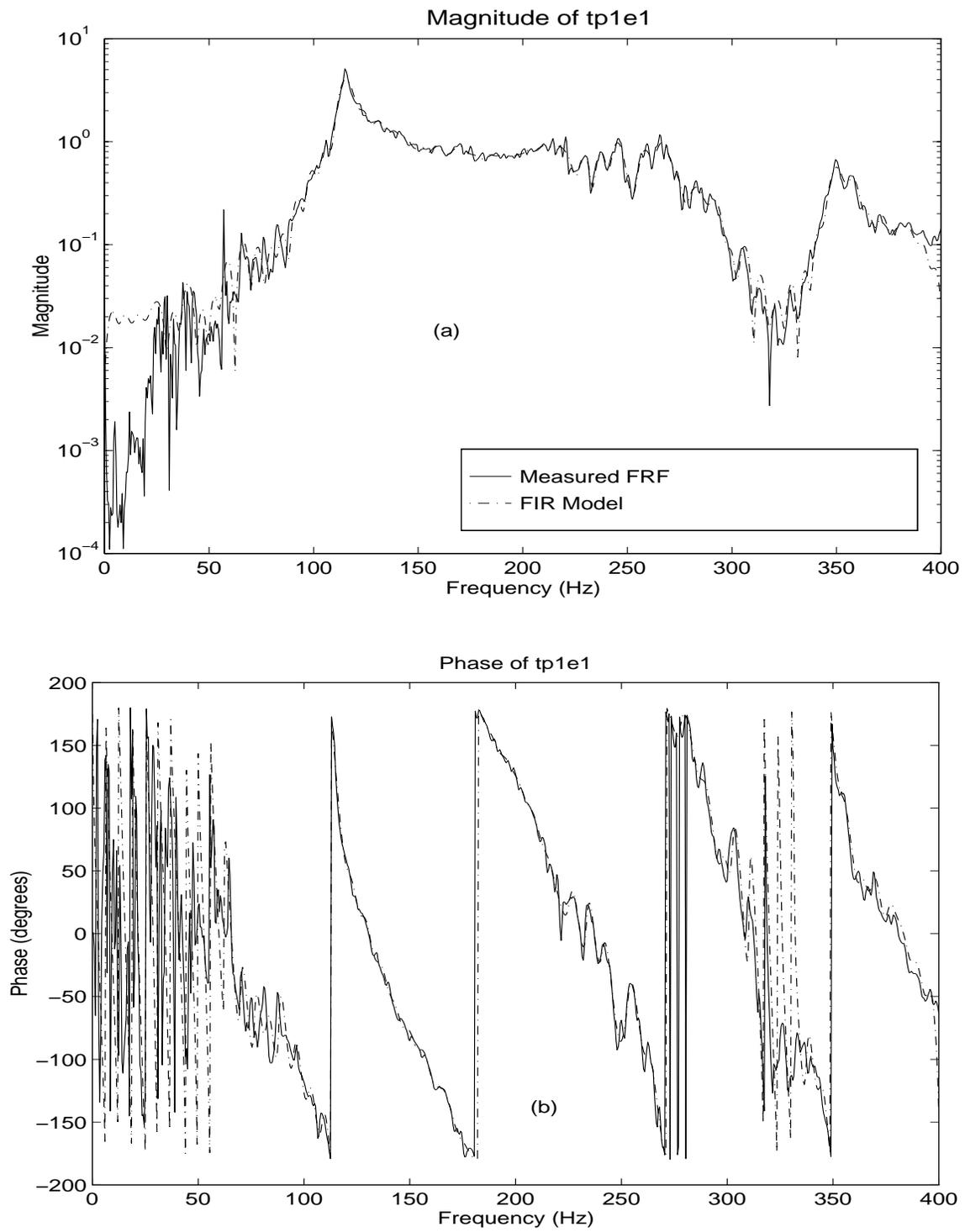
Figure 5.3 Frequency response function and its FIR model for the primary path No.1, a) magnitude response, b) phase response.
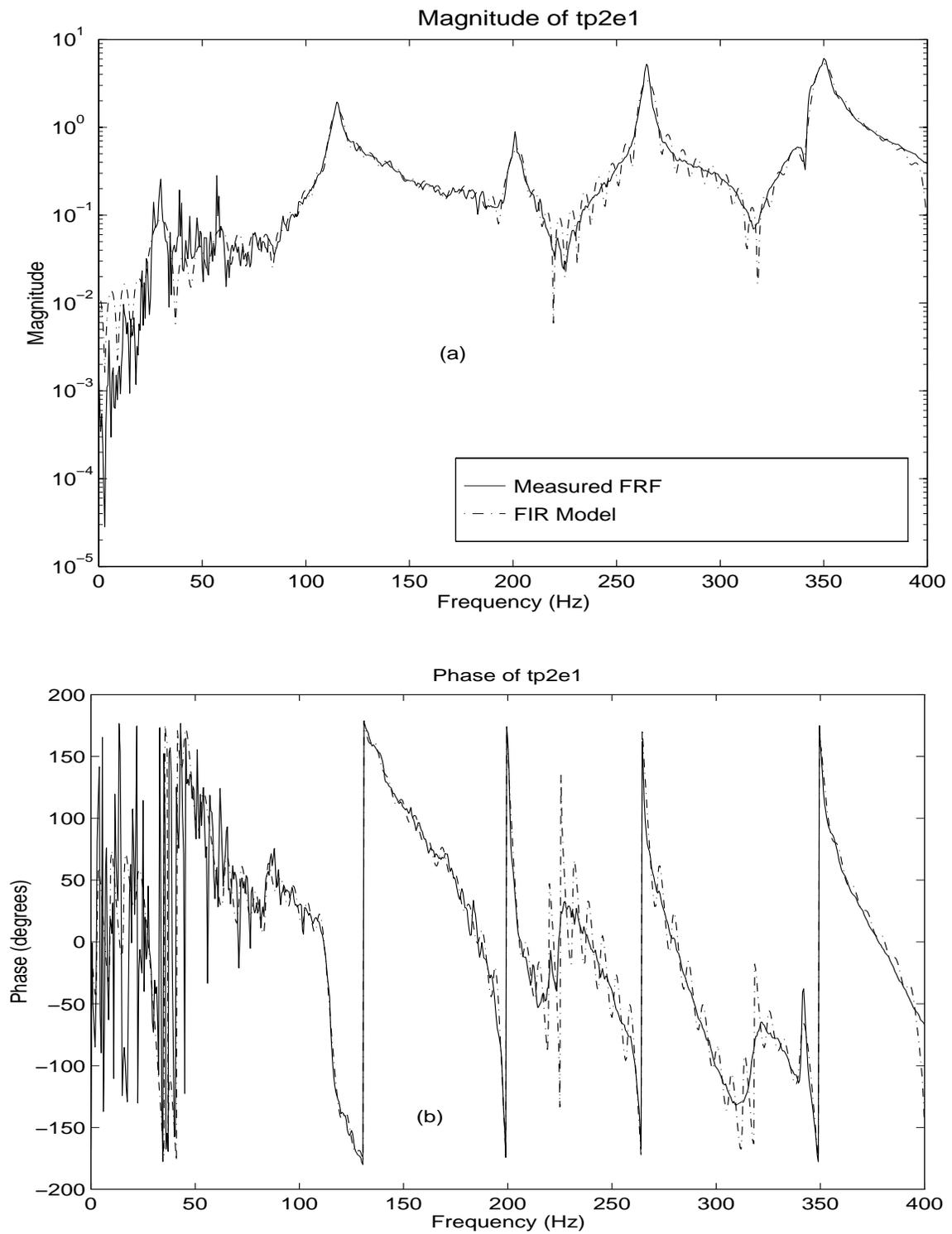
Figure 5.4 Frequency response function and its FIR model for the Primary path No.2, a) magnitude response, b) phase response.
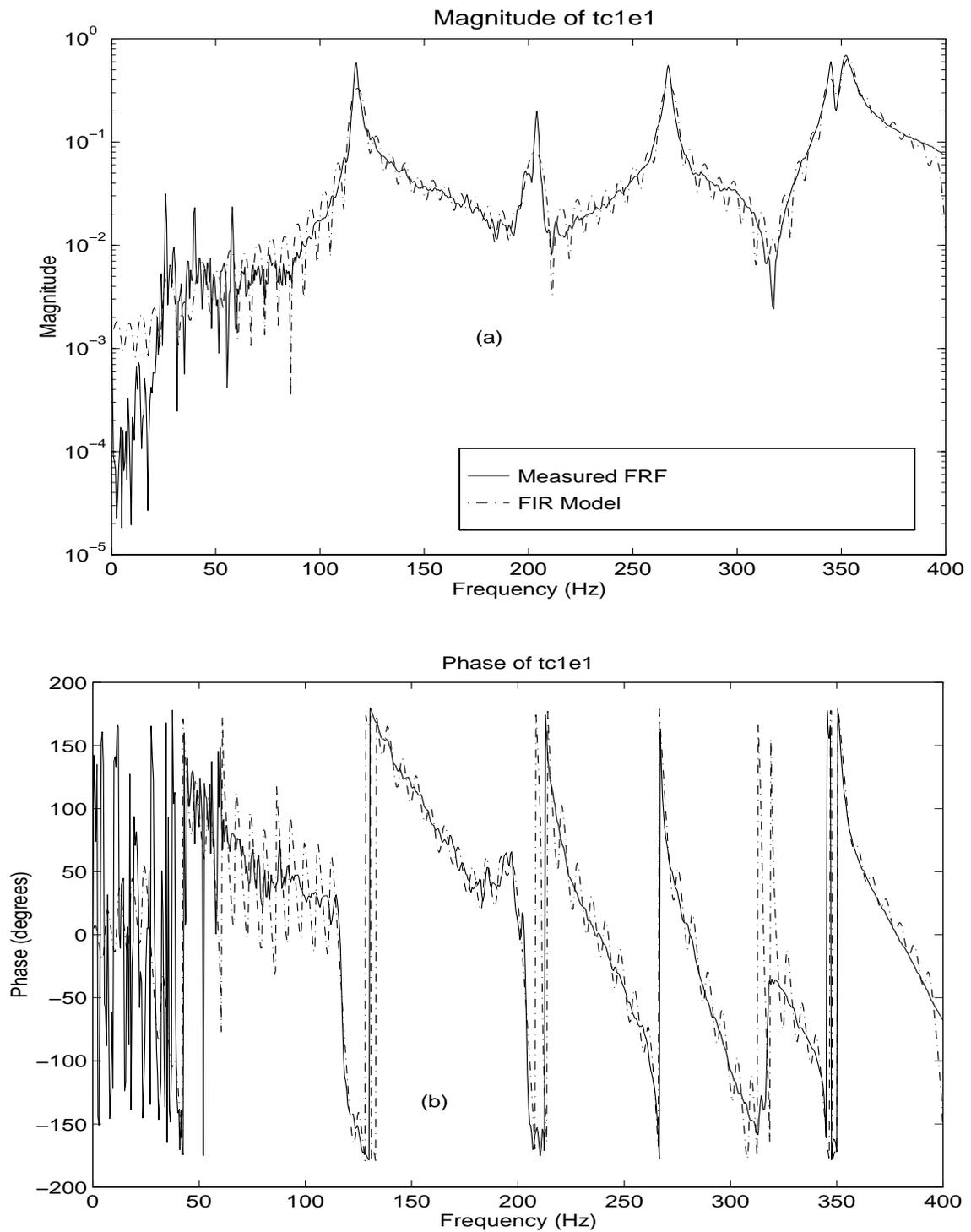
Figure 5.5 Frequency response function and its FIR model for the error path, a) magnitude response, b) phase response.

5.2.2 MRMI versus MRSI

As discussed in section 3.1.5, two configurations (MRSI and MRMI) are usually considered for multiple reference ANC, and their performances are significantly different. This section investigates the performance difference of the two configurations as shown in Figure 5.6, in which two independent random signal generators are used as the noise sources. For the MRMI configuration, the two reference signals are obtained directly from the two random signal generators. Since the reference signals are uncorrelated, there is no preprocessing involved in the simulation. The output signals of the two filters are added together to drive the secondary source. For the MRSI configuration, the reference signal is the summation of the signals from the two random signal generators.

The optimum filter weight vectors for the MRMI configuration can be calculated using the norm equation defined in equation (3.1.29). The expected values inside the matrix **R** and the vector **P** were approximated using the average of 4096 samples. After obtaining the optimum weight vectors through the matrix inversion and multiplication, they are substituted into the controller filters. A sequence of error signal after control is shown in Figure 5.7 (b).

The optimum filter weight vector for the MRSI configuration can be calculated using the norm equation defined in equation (2.2.13). It should be noted that the reference signal is the summation of the two disturbance signals, and there is only one control filter in contrast to the MRMI configuration. The expected values inside the matrix **R** and the vector **P** for the MRSI were similarly approximated using the average of 4096 samples. After obtaining the optimum weight vector, the sequence of the error signal after control was calculated in the same way as that for the configuration MRSI, and the results are shown in Figure 5.7(a).

Figure 5.7 shows that the error signal for the MRMI is much lower than that for the MRSI. This indicates that the MRMI has better performance than the MISI. Although perfect noise cancellation for the MRMI can be achieved theoretically since the system is causal, completely coherent and without any additional noise, the error signal does not reach zero. This is due to the finite filter length effect since each controller filter for the MRMI has only 128 coefficients. Each controller filter for the MRMI has 128 coefficients, while the controller filter for the MRSI has 256 coefficients. The number of coefficients is selected such that the same number of filter coefficients can be implemented in the DSP without exceeding its computational limits when doing real time experiments.

A comparison of the power spectrum is shown in Figure 5.8, which clearly indicates that the performance of the MRMI is much better than that of MRSI. In fact, 10 dB overall noise reduction for the MRMI is achieved, while only about 3 dB overall noise reduction is achieved for the MRSI. It should be noted that the maximum noise reduction occurs in the vicinity of the resonance frequencies, while at the off-resonance frequencies, only a small amount of noise attenuation is achieved.

(a)

(b)
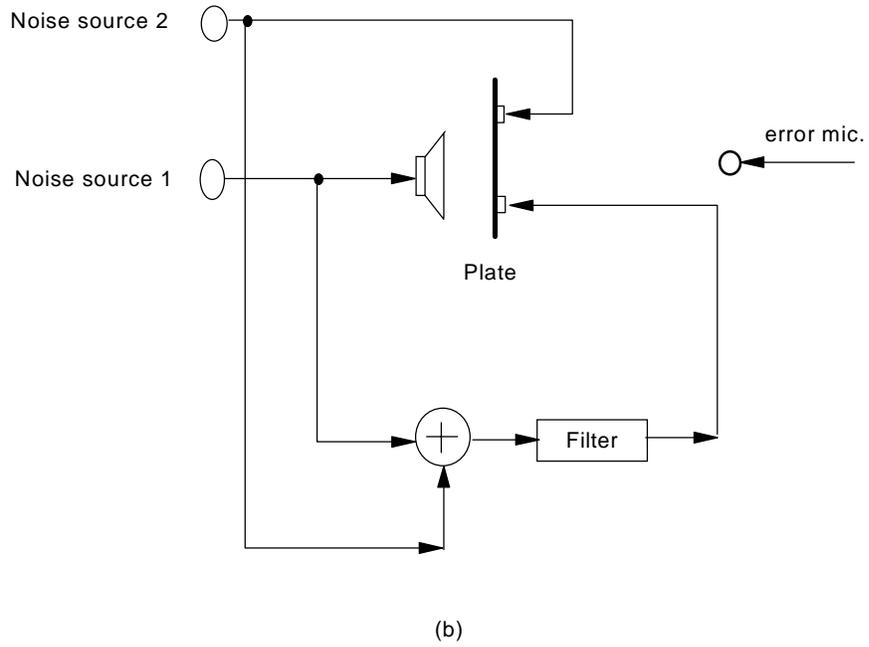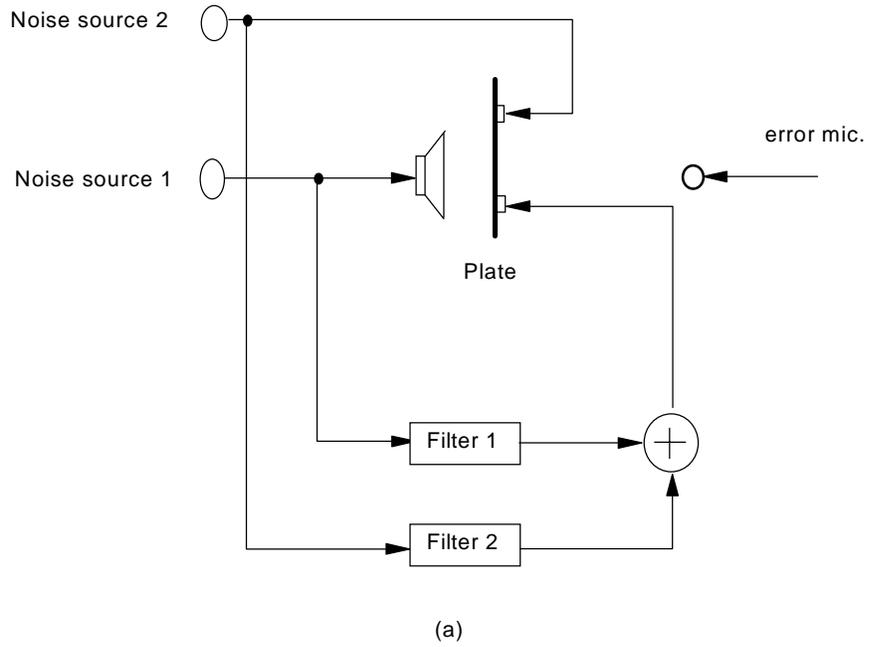
Figure 5.6 Simulation of a multiple reference active noise control system a) MRMI, b) MRSI.
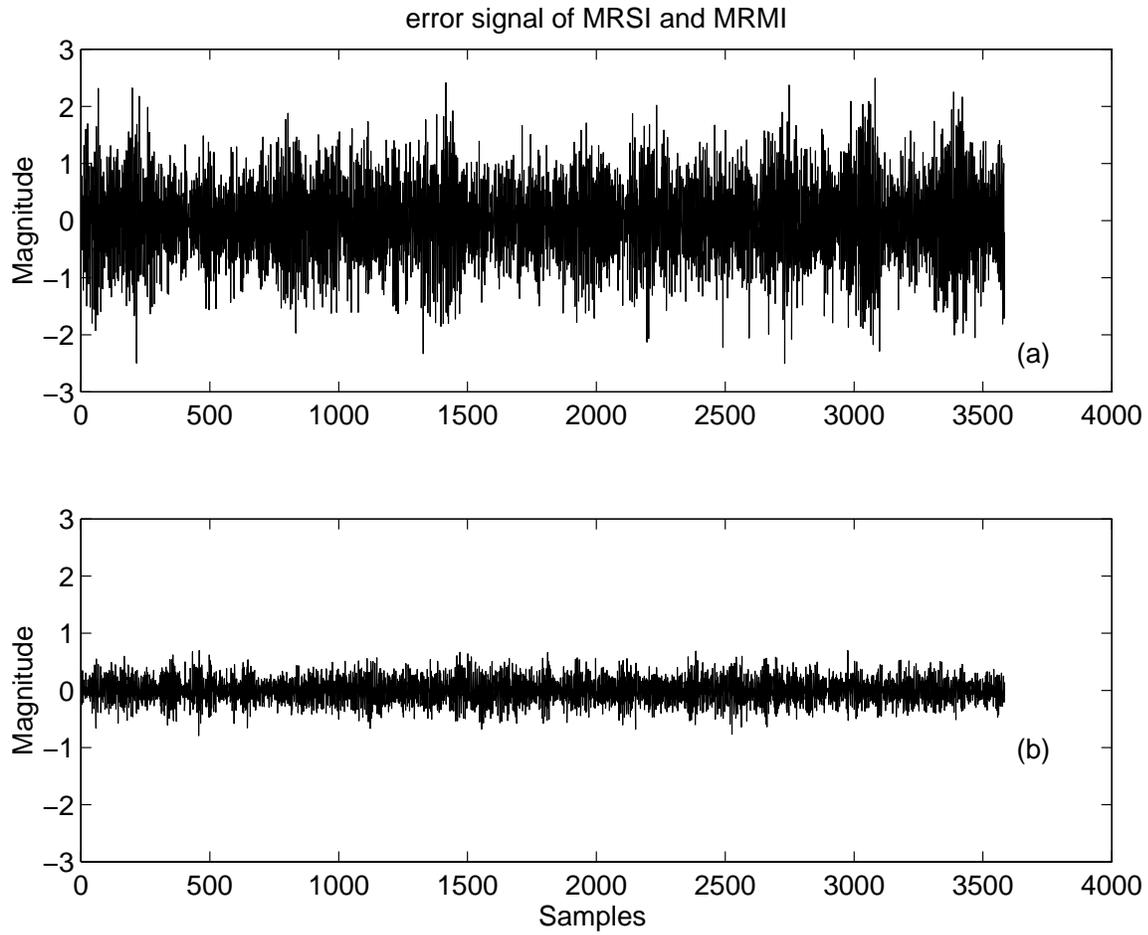
Figure 5.7 Optimum error signals in the time domain, a) multiple reference single input (MRSI), b) multiple reference multiple input (MRMI).
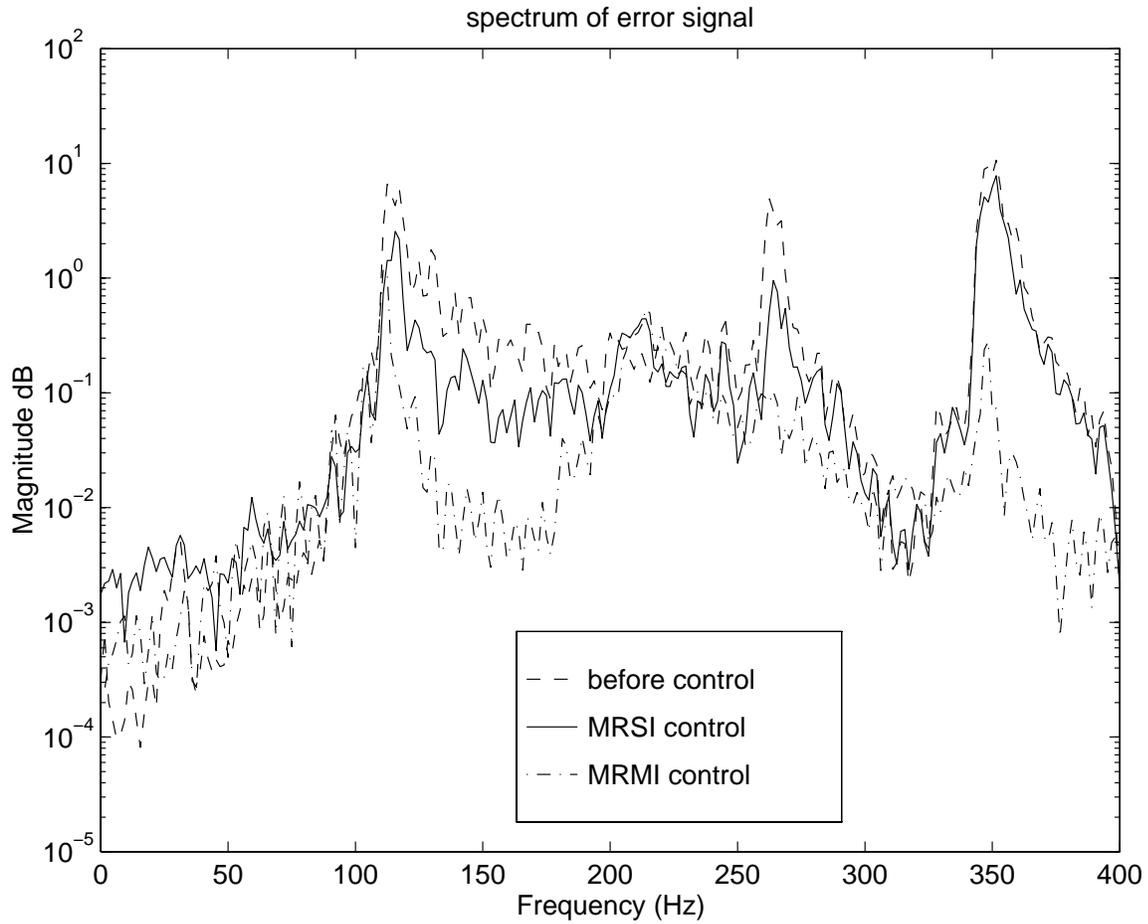
Figure 5.8 Spectrum of the optimum error signals, multiple reference multiple input (MRMI) versus multiple reference single input (MRSI).

5.2.3 Effect of Decorrelation Filters

As mentioned earlier, feedforward active noise control requires that the reference signals be coherent with the error signals. The reference signals are usually measured with various sensors, and each reference sensor may be influenced by more than one noise source. As a result, the reference signals are correlated to some extent regardless of the characteristics of the noise sources.

In this simulation, random signal generators were used to produce two independent noise sources, and the reference signals were obtained indirectly from the two noise sources. The relationship between the reference signals and the noise sources are shown in Fig.5.9. The first reference signal r_1 is exactly the same as the first noise source signal n_1. The second reference signal r_2 is the combination of the second noise source n_2 and the first noise source n_1 filtered through a band-pass filter, that is

$$r_1(k) = n_1(k) \tag{5.2.1}$$
$$r_2(k) = C_0 * n_2(k) + n_1(k)H(Z) \tag{5.2.2}$$

where C_0 is a constant. The two uncorrelated noise source signals, n_1 and n_2, are uniformly distributed between -1 and 1. The cut off frequencies for the band pass filter $H(z)$ are selected to be 160 Hz and 320 Hz. A FIR filter with four coefficients is used here to implement the band pass filter. The windows method is adopted to design the band pass FIR filter and the resultant four coefficients vector is {-1.009, 6.875, 6.875, -1.009}.
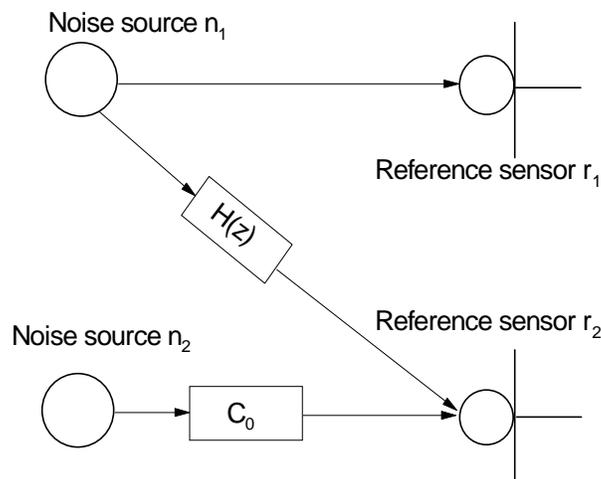
Figure 5.9 Noise sources and reference sensors for the simulations.

The two reference signals in equations (5.3.1) and (5.3.2) are correlated due to the common contributions from noise source n_1, and their correlation function can be varied if a different constant C_0 is selected. Corresponding to each selected constant C_0, computations are carried out to obtain the correlation matrix R for the reference signals, the decorrelated reference signals, the filtered reference signals, and the decorrelated filtered reference signals. Based on the correlation matrix, the corresponding eigenvalue spread is also computed and the results are shown in Table 5.2.

Table 5.2 Eigenvalue spread versus correlation

C_0	Reference signals without D filter	Reference signals with D filter	Filtered reference without D filter	Filtered reference with D filter
1.0	17.7	3.2	8.4×10^6	2.3×10^5
0.5	48.2	3.4	2.9×10^6	3.1×10^5
0.2	242.5	3.0	1.1×10^7	2.2×10^5

It is clear that the eigenvalue spread of the reference signals is smaller after it is processed through the decorrelation filter. In fact, the convergence speed is determined by the filtered reference signals instead of the reference signals. Thus, in order to improve the convergence speed, the eigenvalue spread for the filtered reference signals must get smaller as well. This requirement is indeed satisfied since, although decorrelation is only applied to the reference signals, the correlation between filtered reference signals is also affected. It is also interesting to note the eigenvalue spread for the filtered reference signals is much larger than that for the reference signals. This is due to the stretching effect of the error path.
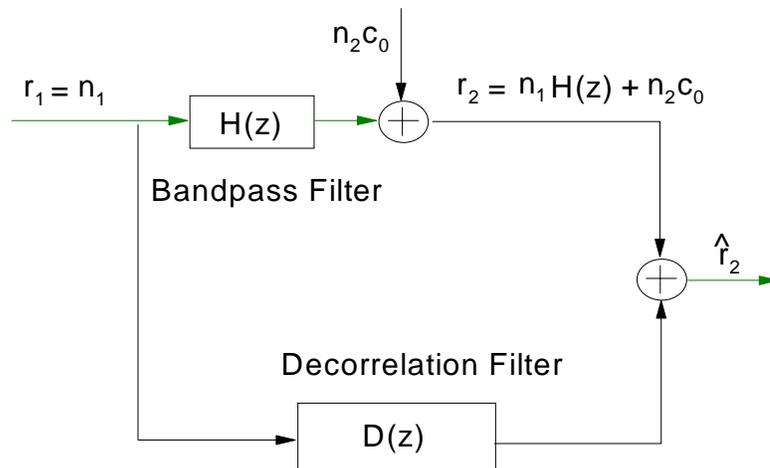
Figure 5.10 Input and output configuration for the decorrelation in the simulation.

In this simulation, the decorrelation filter is implemented with a fixed Wiener filter instead of an adaptive filter since the correlation between the reference signals is time invariant. The input and output configuration of the decorrelation filter is shown in Figure 5.10. The requirement for the decorrelation filter is to remove the component in the reference signal r_2 which is correlated with the reference signal r_1. Such a requirement is essentially a system modeling problem if the band-pass filter is considered as a plant. In theory [43], the band pass filter can be perfectly modeled with the decorrelation filter since the plant disturbance due to noise source n_2 is uncorrelated with noise source n_1. The desired decorrelation filter weight vector can be calculated using equation (2.2.13), in which the filtered reference signal $\hat{x}$ and the desired signal d are replaced by r_1 and r_2 respectively. Sample averaging is used to estimate the expected value and the weight vector is calculated as {1.025, -6.888, -6.863, 1.003}.

When the constant C_0 is zero, the two reference signals are completely correlated. It follows that the second reference signal after decorrelation is almost zero. In this case there is no need to use a multiple reference control configuration, since a single reference control configuration should perform equally as well.

The optimum error spectrums with and without the decorrelation filter were obtained when the controllers are operated as the fixed Wiener filters. The results indicate that the optimum error signals are almost the same regardless of whether or not the decorrelation filter is used, as shown in Figure 5.11. This result is expected, since although the decorrelation filter changes the eigenvalue spread of correlation matrix of the reference signals, it does not improve the coherence of the system.

In order to examine the effect of the decorrelation filter on improving the convergence speed, the conventional FXLMS algorithm discussed in section 2.3.1 was first applied. After letting the controller weight vectors converge for 30 seconds, the convergence process was frozen to obtain a set of partially converged weight vectors. Then, these weight vectors are used to compute the spectrum of the error signal. In the same way, the spectrum of the error signal for the DFXLMS algorithm discussed in section 4.1 was also computed. The results are shown in Figure 5.12. After 30 seconds, 4.6 dB noise reduction was achieved without the decorrelation filter, while 6.2 dB noise reduction was achieved with the decorrelation filter. This means that the convergence speed with the Decorrelated FXLMS algorithm is faster than the conventional FXLMS algorithm, since the eigenvalue spread of the system is smaller after the decorrelation of the reference signals, as indicated in Table 5.2.

The learning curves for both the DFXLMS algorithm and traditional FXLMS algorithm are shown in Figure 5.13. The results shows that the convergence speed of the DFXLMS algorithm is about 3 times as fast as that of the traditional FXLMS algorithm, and the improvement of the convergence speed is comparable in magnitude to the improvement of the corresponding eigenvalue spread as shown in Table 5.2. It is very interesting to observe that, after 6 minutes of convergence, the power of the error signal without the decorrelation filter is still larger that that with the decorrelation filter. Such a result does not indicate that the decorrelation filter improves the mean square error of the FXLMS algorithm. In fact, the mean

square error stays the same since the optimum mean square error does not change with the decorrelation filter as shown in Fig. 5.11, and the misadjustment stays the same after the preprocessing through the decorrelation filter. The disparity between the mean square error after 6 minutes is due to the very long convergence time for both algorithms because the eigenvalue spread for the filtered reference signals is extremely large. If sufficient time is given for the two algorithms, it is expected that the mean square errors will approach the same value.
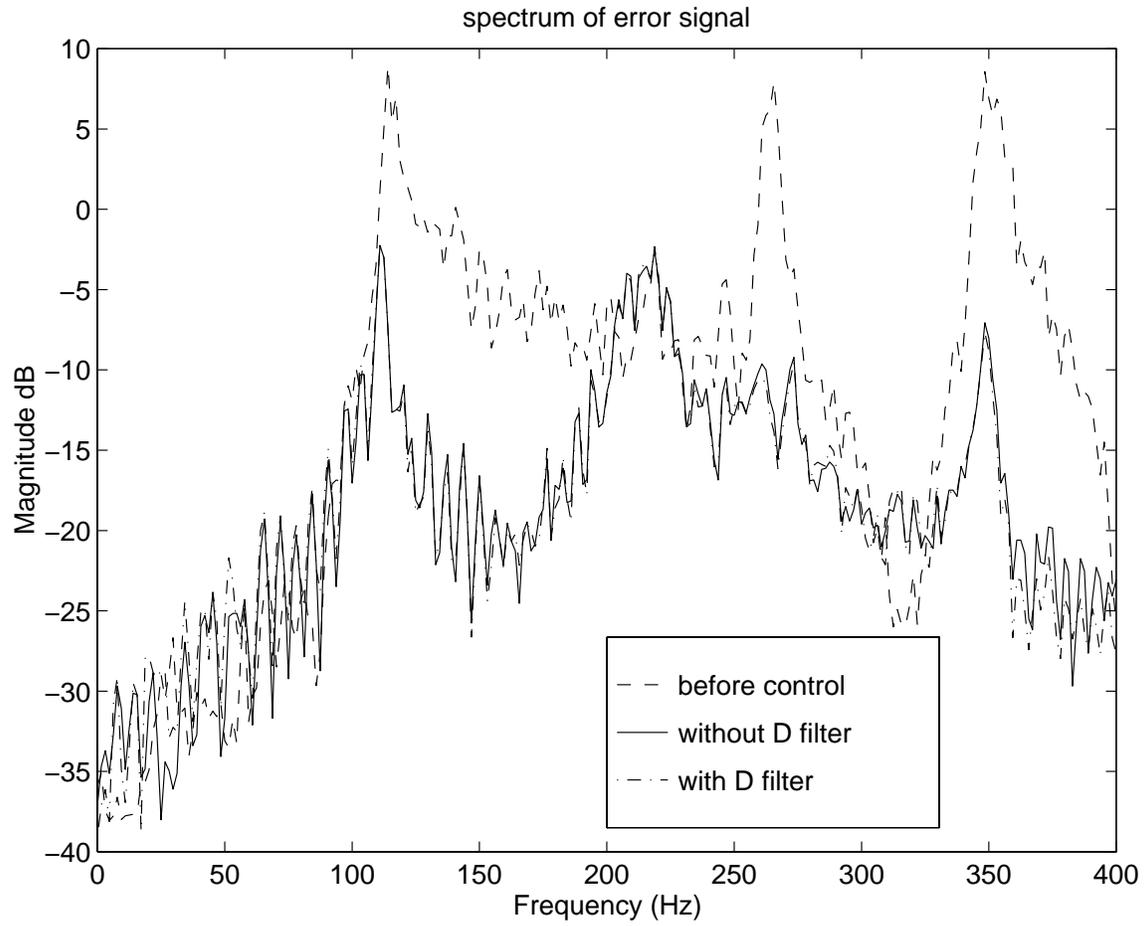
Figure 5.11 Comparison of the optimum error spectrum, DFXLMS algorithm versus FXLMS algorithm.
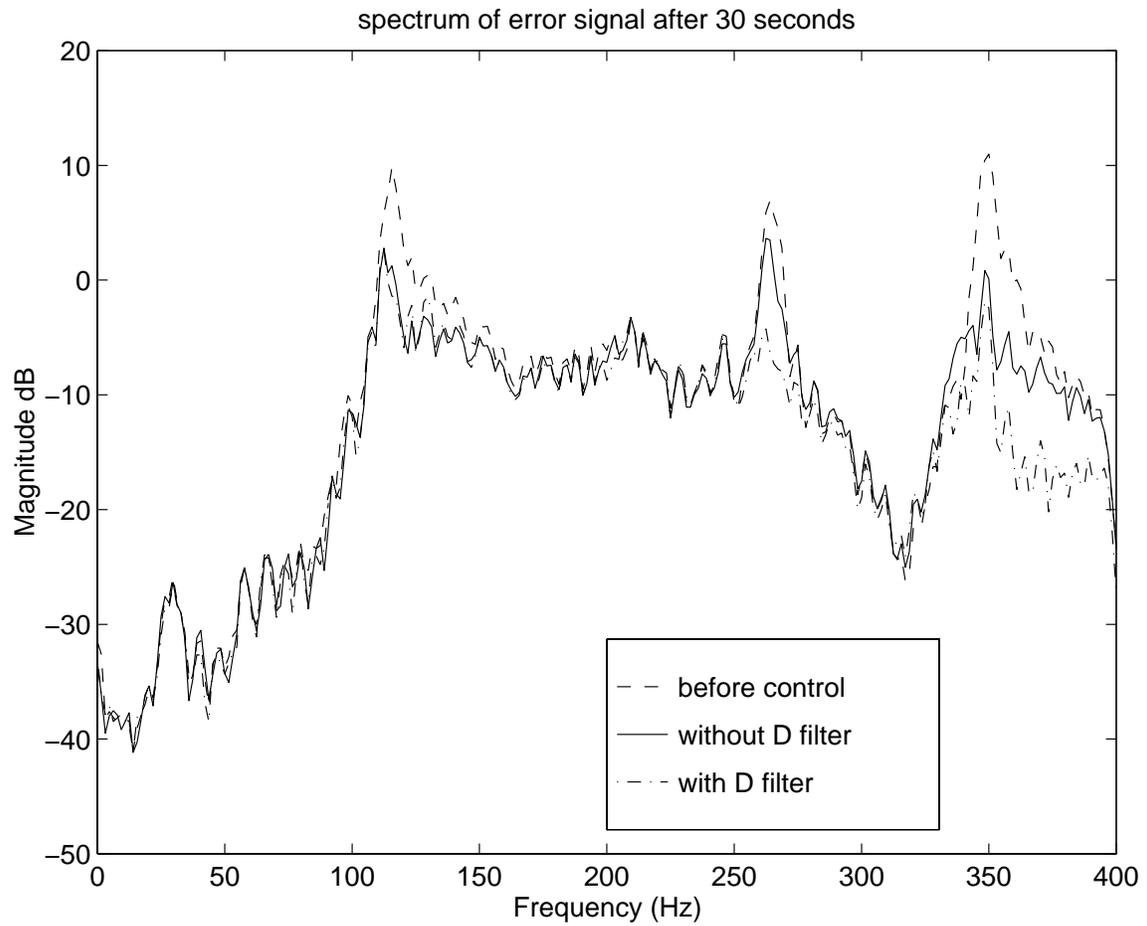
Figure 5.12 Comparison of the error signal after 30 seconds of convergence time, FXLMS algorithm versus DFXLMS algorithm.
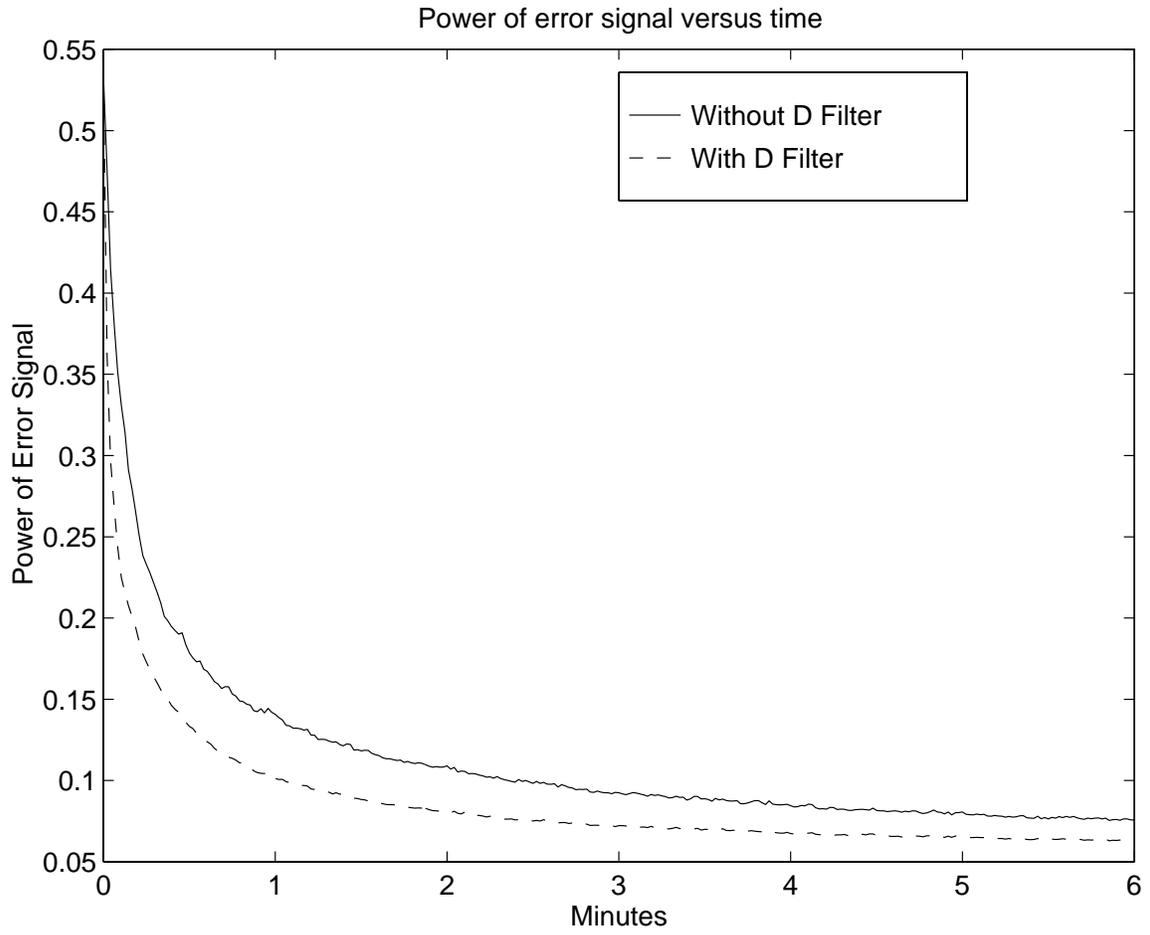
Figure 5.13 Learning curves of the DFXLMS algorithm and FXLMS algorithm.

5.3 Experimental Results

In order to examine the practical issues associated with multiple reference ANC, experiments were conducted, and the results are presented in this section. In contrast to the simulations discussed in section 5.3, the experiments must be conducted in real time, which means that extensive computations have to be carried out during each sampling interval such that the control output signals can be generated to drive the secondary source. This computational requirement implies the use of DSP.

For the experiments, the controller is based on adaptive FIR filters, and the number of coefficients for each FIR filter is selected to be 128. The error path between the secondary source (PZT #2) and the error sensor is also modeled with a FIR filter, and the same number of coefficients is used. The frequency range of the noise field is selected to be below 400 Hz. These parameters are selected based on the computational limit of the DSP, since increasing frequency range generally requires more filter coefficients to get satisfactory results, which in turn requires more computations.

The test rig has been discussed in section 5.1, and the block diagram showing the various elements for the experiments is presented in Figure 5.14. The heart of the system is the TMS320C30® DSP board, which is used to implement the algorithms. The A/D and D/A conversions are carried out through two additional I/O boards, which provide 32 input channels and 16 output channels. The DSP board along with two associated I/O boards is plugged into a PC. A graphical user interface (GUI) running under the host PC is provided to adjust various control parameters and display DSP data. Since the disturbance signals generated within the DSP are digital in nature, they are transformed into analog signals through D/A converters to drive the primary noise sources (speaker and PZT #1). Similarly, the control signal is also transformed into analog signal through D/A converters to drive the secondary source (PZT #2). Since the frequency range for the experiments is selected to be below 400 Hz, all the signals are low-pass filtered so that the frequency components above 400 Hz are negligible. In addition, since the signals generated within the DSP have very small power, in order to drive the speaker and PZTs, they are also fed into power amplifiers. It should be noted that the signal from the error microphone is fed into a high-pass filter to eliminate the dc signal drift.

In the experiments, the disturbance signals were generated through the DSP and were directly used as the reference signals. As a result, the reference signals are perfect with unity coherence function and no feedback from the secondary source. This should be contrasted to real applications, in which the reference signals are usually measured through various types of sensors. The use of reference sensors usually results in measurement noise and feedback signals. Thus, in real applications, the selection of the positions for the reference sensors is very important in order to get good coherence and minimal feedback. In addition, the causality problem may be avoided if some signal delays are added in the primary path through the DSP buffers.

The controllability and observability of each noise source are first examined in the experiments. With only the acoustical disturbance being excited, 7.8 dB overall noise reduction was obtained as shown in Figure 5.15. While, with only the structural disturbance being excited, 12.6 dB overall noise reduction was achieved as shown in Figure 5.16. It is interesting to note that, in Figure 5.15, the (1,1) mode is the most dominant mode observed with the error microphone, while, in Figure 5.16, all the modes except the (2,1) mode are equally observed. It should also be noted that the noise reduction at the resonance frequencies is much greater than that at the off-resonance frequencies.

As for the simulation shown in Section 5.2, the performance of the two different configurations (MRSI and MRMI) as well as the convergence speed of the two algorithms (FXLMS and DFXLMS) are compared. The similar experimental results are presented in the following subsections.
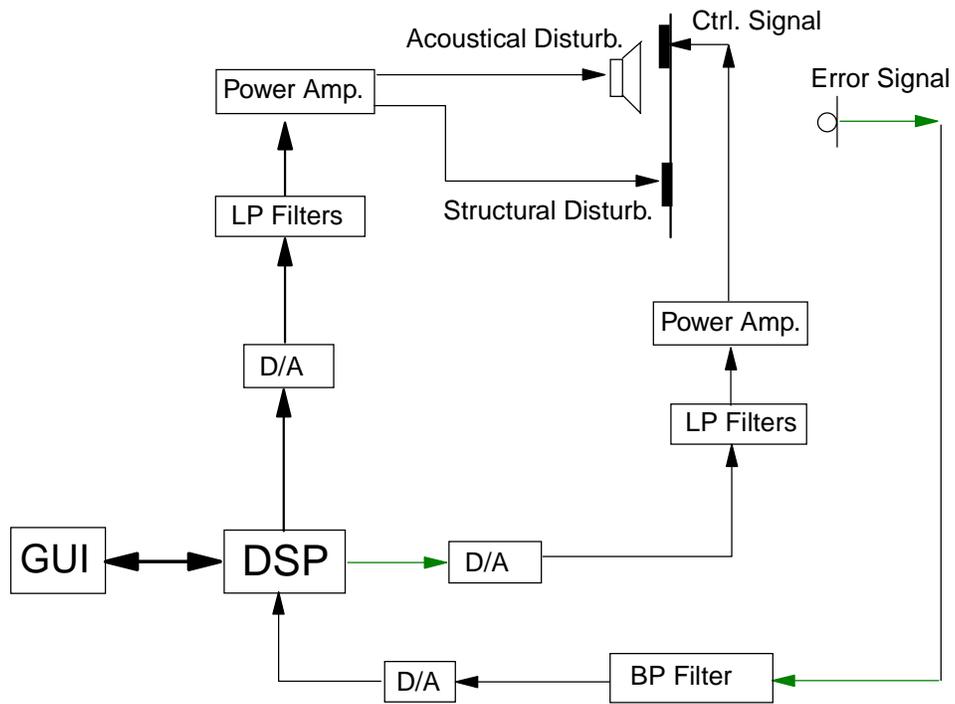
Figure 5.14 Block diagram of the various elements for the experiments.
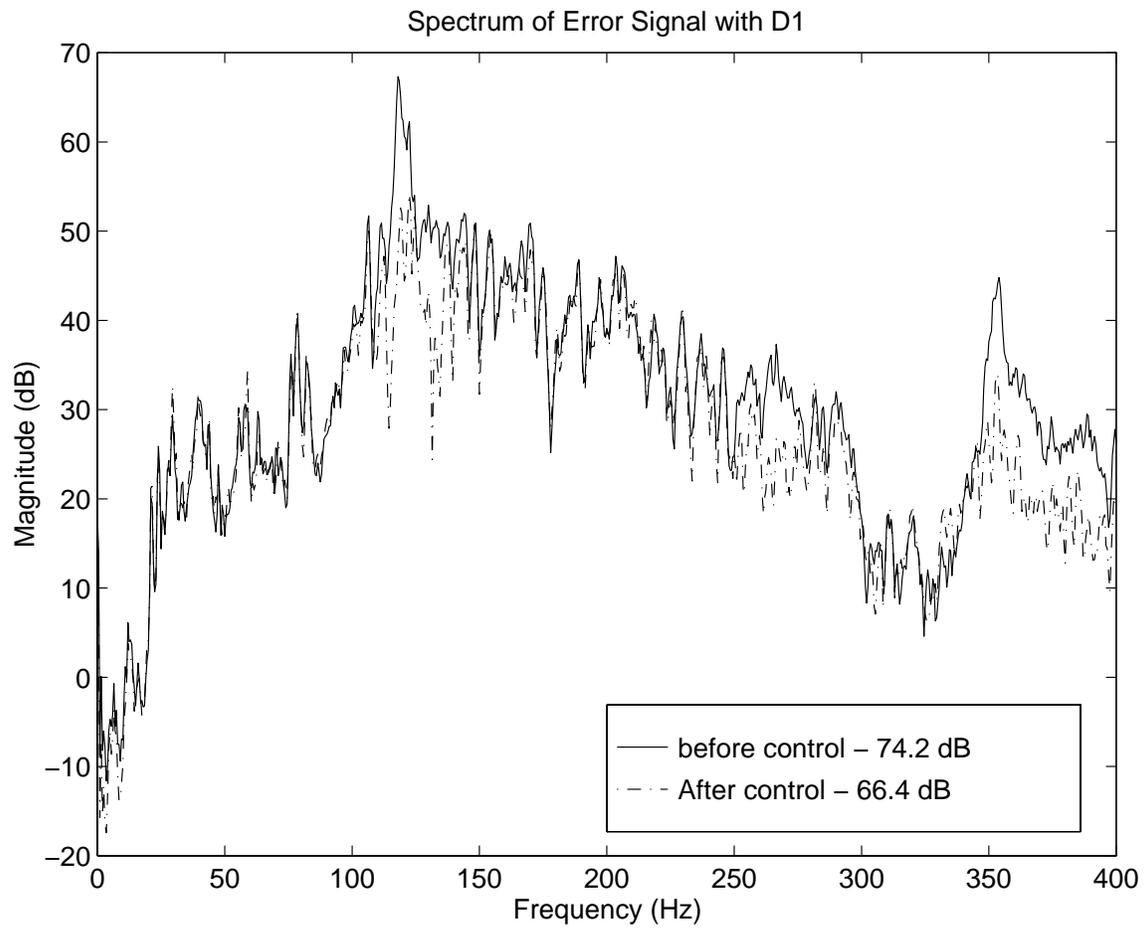
Figure 5.15 Spectrum of error signal with only disturbance No.1 (Speaker).
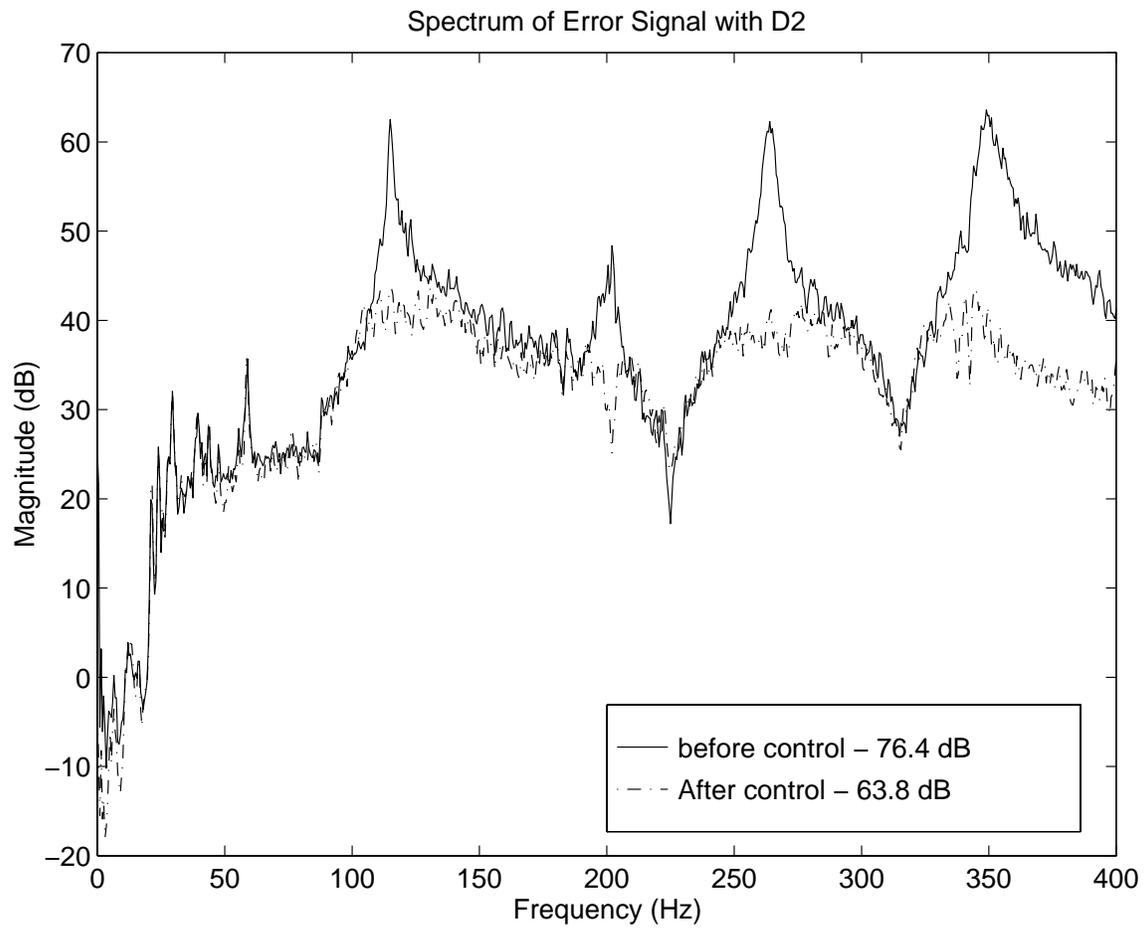
Figure 5.16 Spectrum of error signal with only disturbance No.2 (PZT #1).

5.3.1 MRMI versus MRSI

As discussed in Section 3.1.5, when there are multiple noise sources with different paths to the error sensors, the MRMI configuration is needed to achieve satisfactory noise reduction. Although the MRSI configuration simplifies the ANC system, the noise control effect is generally compromised.

In this experiment, two uncorrelated disturbance signals were generated with random signal generators to drive the primary sources for both configurations. For the MRMI configuration, each reference signal was obtained directly from each disturbance signal, and no decorrelation filter was needed since the reference signals were uncorrelated. For the MRSI configuration, the reference signal was obtained by summing the two disturbance signals together.

A comparison of the power spectrum of the error signal before and after control using the two configurations is shown in Figure 5.18. The results were based on the FXLMS algorithm, whereas the similar results obtained in the simulations were based on the fixed Wiener filters. Only 2.3 dB noise reduction with the MRSI configuration was achieved, while, overall 11.6 dB noise reduction was achieved with the MRMI configuration. The poor performance of the MRSI configuration is expected since the error signal is due to multiple noise sources passing through multiple different paths as explained in Section 3.1.5. The performance difference between two configurations can also be explained through coherence analysis.

Measured with a B&K 2032 signal analyzer, the ordinary coherence functions between the reference signals and the error signal are shown in Figures 5.18 and 5.19. Since the two coherence functions for the MRMI configuration are supplementary throughout the frequency range, it is therefore possible to achieve good overall noise reduction. However, the coherence function for the MRSI configuration is very poor for most of the frequency range, thus, it is impossible to achieve good noise reduction.
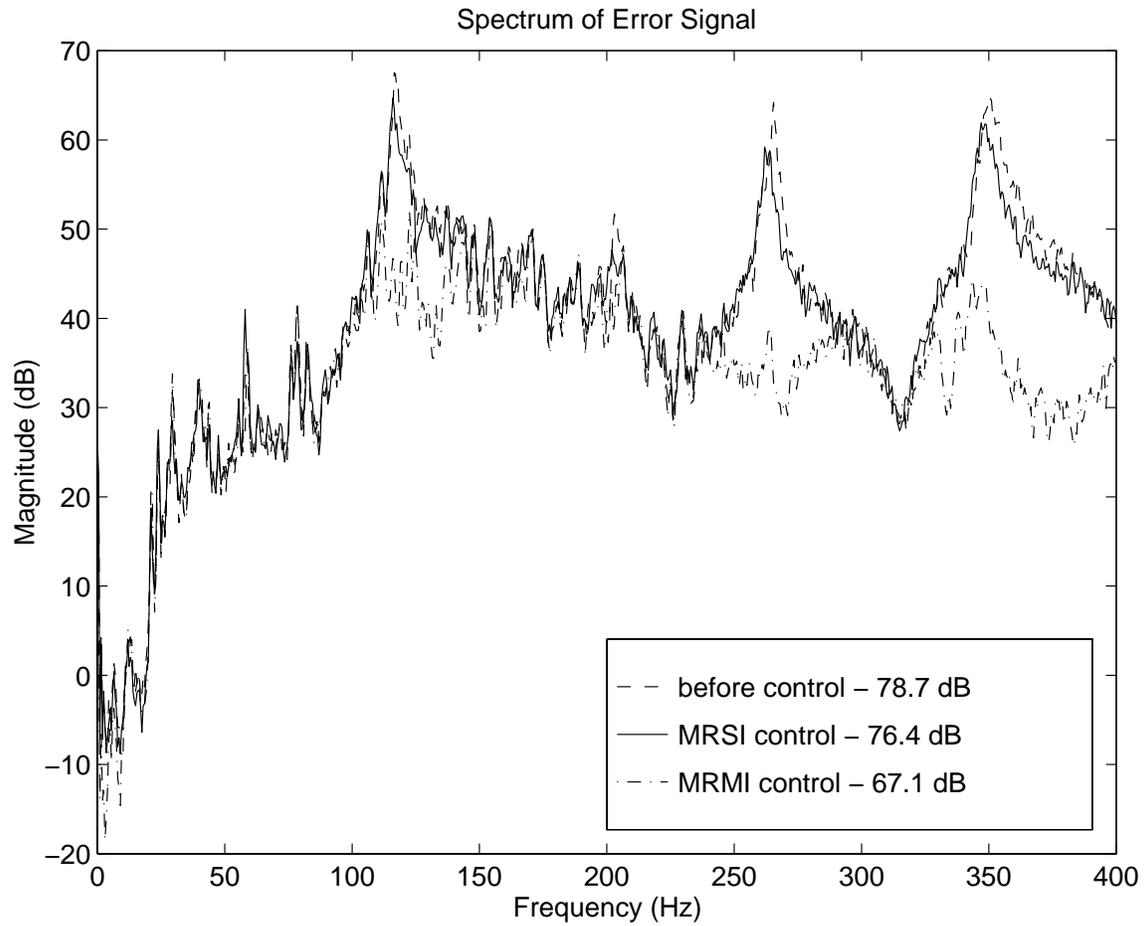
Figure 5.17 Spectrum of the steady state error signals, multiple reference multiple input (MRMI) versus multiple reference single input (MRSI).
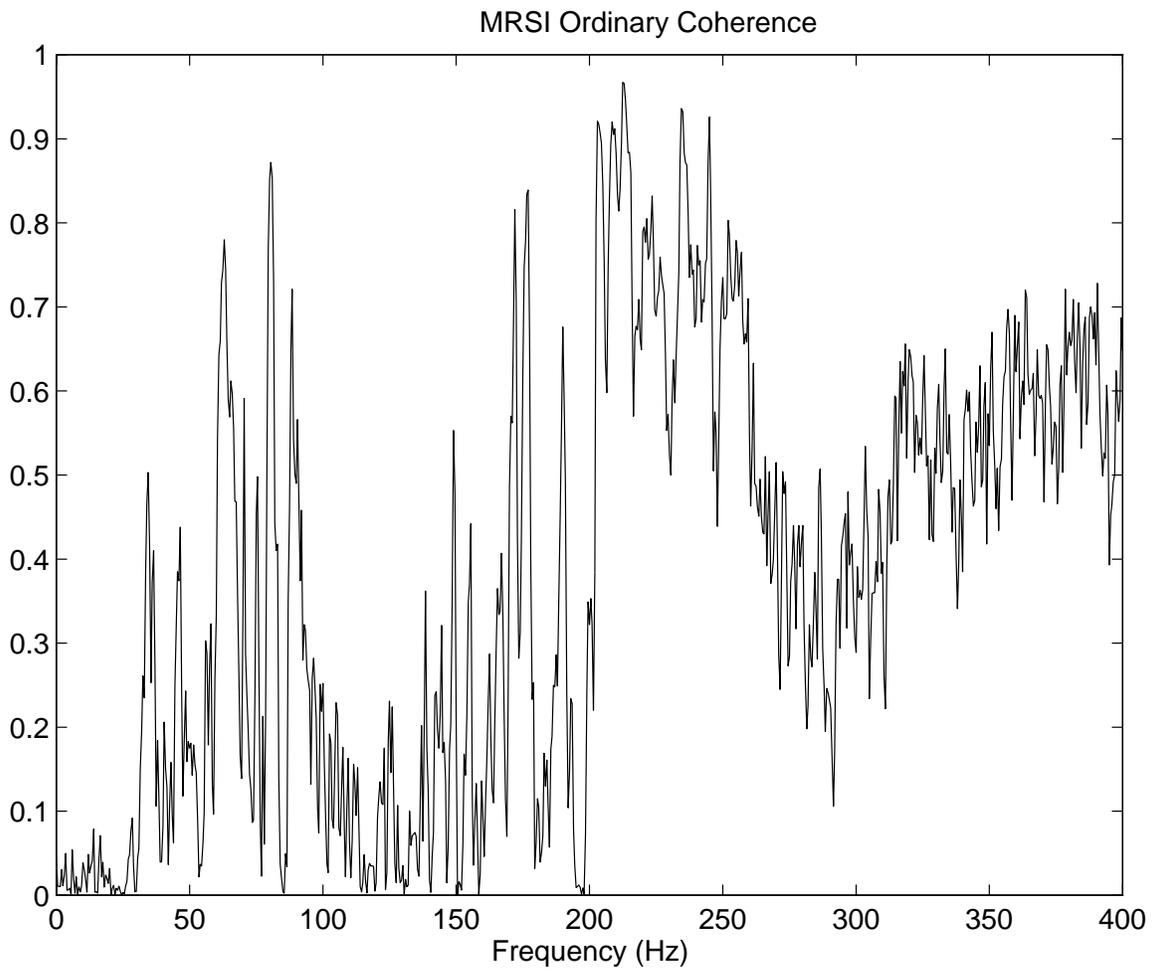
Figure 5.18 Ordinary coherence functions between the reference signal and the error signal with multiple reference single input (MRSI) configuration.
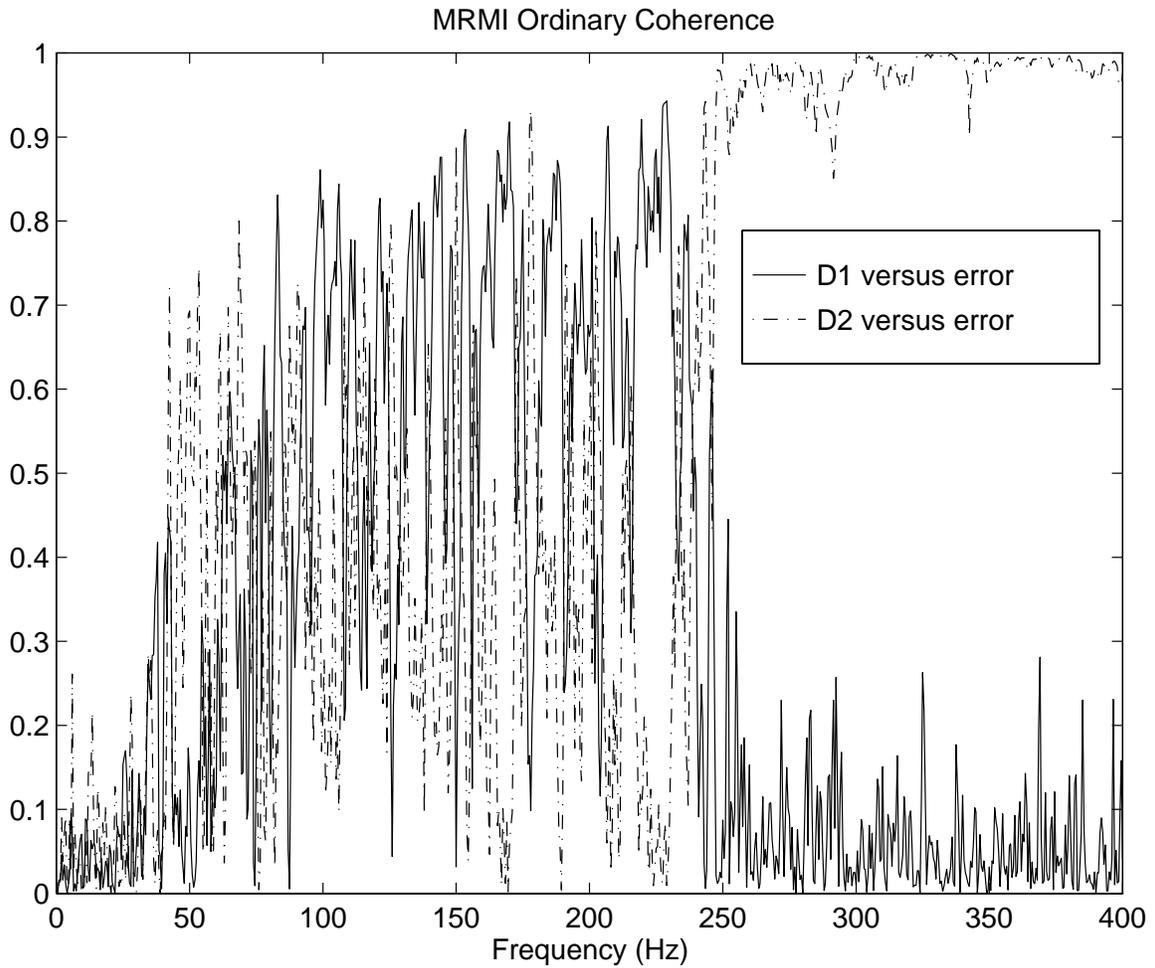
Figure 5.19 Ordinary coherence functions between the reference signals and the error signal with multiple reference multiple input (MRMI) configuration.

5.3.2 Effect of Decorrelation Filters

The effectiveness of the decorrelation filter to improve the convergence speed has been demonstrated in the simulation discussed in Section 5.3.2. The effect of the decorrelation filter is now experimentally investigated. Except otherwise stated, all the parameters in these experiments are the same as those used in the corresponding simulation.

The conventional FXLMS algorithm and the DFXLMS algorithm were applied, the auto-spectrum of the corresponding error signals was measured as shown in Figure 5.20. Both auto-spectrums were obtained after 30 seconds of convergence time, and the convergence parameter μ was chosen to be 0.01 for both control algorithms. The results indicate that 9.0 dB noise reduction was achieved with the DFXLMS algorithm, while only 5.3 dB noise reduction with the conventional FXLMS algorithm. A smaller convergence parameter (μ=0.001) was also used, and the results are shown in Figure 5.20. Although both algorithms converged slower, the same tendency showing that the DFXLMS can achieve larger noise reduction was observed. As a conclusion, the DFXLMS algorithm converges faster than the conventional FXLMS algorithm since the eigenvalue spread of the filtered reference signals is smaller after the decorrelation processing, as shown in Table 5.2.

In order to measure the learning curve, the convergence processes of the FXLMS algorithm and the DFXLMS algorithm were started, and the error signal power at 5, 10, 15, 25 … and 360 seconds were measured. These values representing the power of the error signal form the learning curves as shown in Figure 5.22. As in the simulation, the results indicate that the Decorrelated FXLMS algorithm converges about three times faster than the FXLMS algorithm. It is interesting to note that the error signal power for the Decorrelated FXLMS is still larger than that for the FXLMS algorithm after 360 seconds. This, however, does not imply that the Decorrelated FXLMS algorithm can improve the steady state mean square error. In fact, the steady state mean square errors for both algorithms are the same, as discussed in the simulation. Since the eigenvalue spread for the filtered reference signals are extremely large, excessively long convergence time is required for the error signal to reach the steady state.
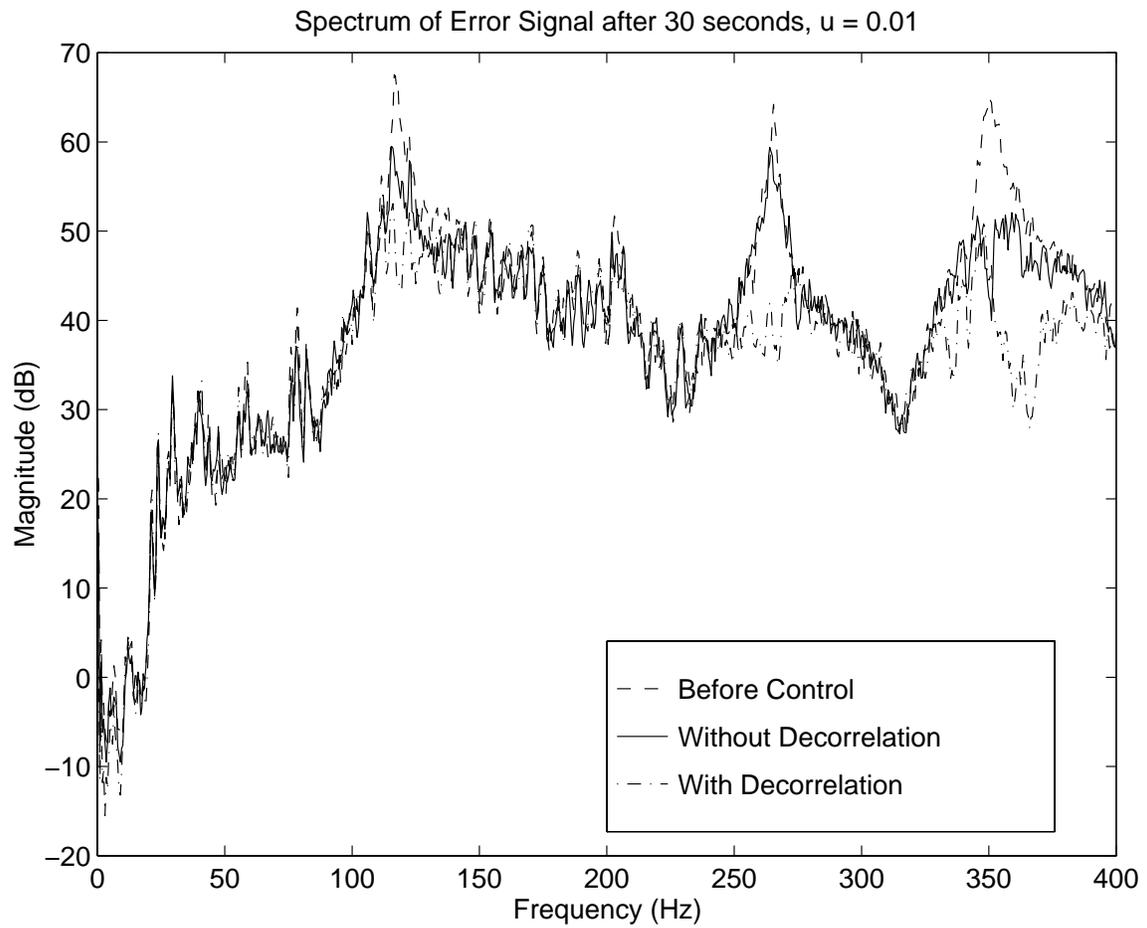
Figure 5.20 Comparison of the spectrum of the error signal after 30 seconds μ=0.01, with decorrelation filters versus without decorrelation filters.
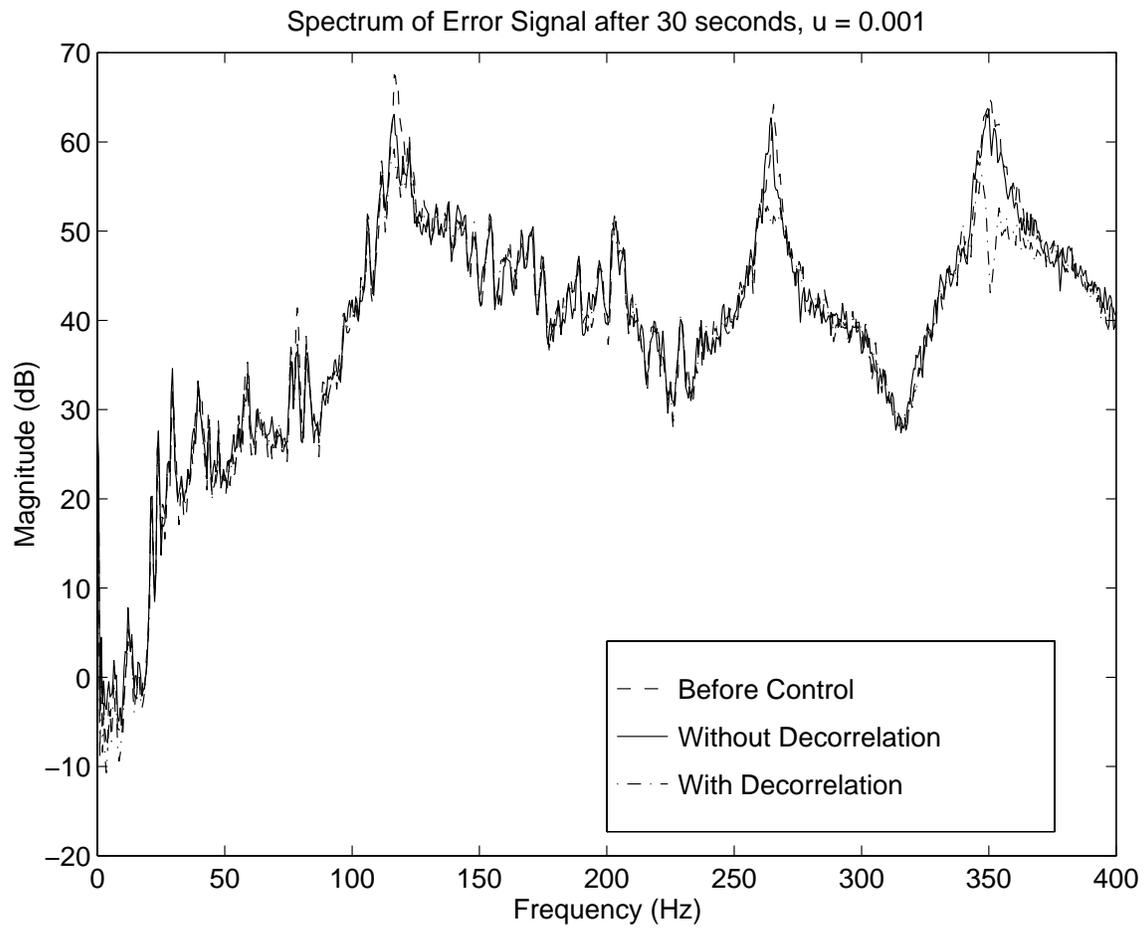
Figure 5.21 Comparison of the spectrum of the error signal after 30 seconds μ=0.001, with decorrelation filters versus without decorrelation filters.
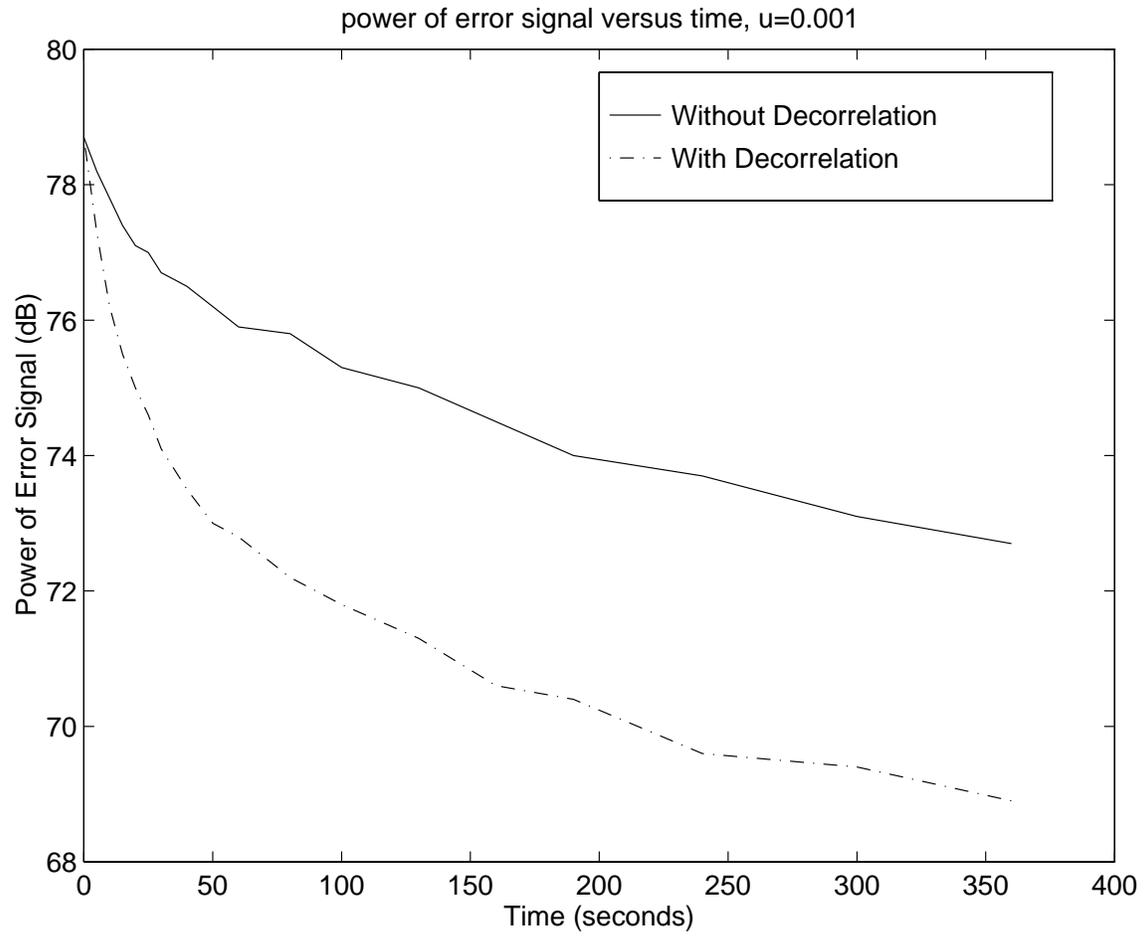
Figure 5.22 Comparison of the error signal power, with decorrelation filter versus without decorrelation filter.

Chapter 6 Conclusions and Future Works

6.1 Conclusions

The behavior of multiple reference broadband active noise control has been studied in both the frequency and time domain, it was particularly noted that if multiple reference signals are correlated, the corresponding system will be ill-conditioned, which results in slow convergence speed and high sensitivity to measurement error.

An approach for decorrelating reference signals has been discussed based on the Wiener filter theory and Gram-Schmidt orthogonalization theorem. Furthermore, the multiple reference DFXLMS algorithm was developed to handle the situation where there are multiple correlated reference signals. The DFXLMS algorithm differs from the traditional FXLMS in that the reference signals are preprocessed through a set of decorrelation filters.

The multiple reference single input (MRSI) configuration has been compared with the multiple reference multiple input (MRMI) configuration. It was noted that when there are several independent noise sources, the MRMI configuration performs much better than the MRSI configuration. The two configurations can achieve comparable results only when the primary paths are similar.

The above conclusions has been demonstrated through simulations and experiments on a plate. The experiments indicate that the DFXLMS algorithm could not only increase the convergence speed, but reduce the noise level as well.

There are more computations involved in the DFXLMS algorithm than in the traditional FXLMS algorithm. The computational cost increases with the number of reference signals and the number of coefficients for each decorrelation filter. In practice, whether to apply the DFXLMS algorithm or the FXLMS depends on the quality of the reference signals and the DSP speed.

6.2 Future Works

Multiple reference active noise control is mainly for complex system, in which there are multiple noise sources. The characteristics of the noise sources have significant impact on the quality of the reference signals. On the other hand, the reference signals are usually obtained through various types of reference sensors. Therefore, the positions and the number of reference sensors also determines the characteristics of the reference signals. As a result, it is highly suggested that the optimization of the reference sensors be studied. The optimization should include both the position and the number of reference sensors. In some real applications when

the exact number of noise sources is not identifiable, the position and the number of reference sensors are usually coupled in the optimization process, and the position of the reference sensors has crucial impact on the number of sensors needed. Therefore, the basic optimization criterion could be achieving maximum noise reduction with minimum reference sensors while keeping the reference signals as uncorrelated as possible.

For a complex system, the computations involved in the multiple reference active noise control are extremely intensive, especially when multiple channels are applied to achieve global noise control effect. This computational requirement suggests the study of efficient algorithms such as frequency domain block algorithm. Furthermore, it is suggested to explore other algorithms to implement more efficiently when multiple reference signals and multiple channels are involved.

There are many other adaptive algorithms for the ANC, which deliver diverse performance due to the different cost functions or the different filter structures. It is possible that some algorithms are less sensitive to the eigenvalue spread of the ANC system. Therefore, it is suggested that the behavior of other algorithms using correlated reference signals be studied and optimized.

Bibliography

[1] P. Leug, "Process of Silencing Sound Oscillations," U.S. Patent No. 2,043,416, 1936.

[2] H. F. Olson and E.G.May, "Electronic sound absorber," J. Acoust. Soc. Am., 28 966-972, Sept. 1956.

[3] L. J. Eriksson and M. C. Allie, "Correlated active attenuation system with error and correction signal input, " U.S. Patent 5,206,911, Apr. 27, 1993

[4] L. J. Eriksson, "Active sound attenuation system with on-line adaptive feedback cancellation," U.S. Patent 4,677,677, June 30, 1987.

[5] J. S. Vipperman, R. A. Burdisso, and C. R. Fuller, "Active control of broadband structural vibration using the lms adaptive algorithm," Journal of Sound and Vibration (1993) 166(2), 283-299.

[6] S. Haykin, Adaptive Filter Theory, 2nd Ed., Prentice-Hall, Englewood Cliffs, NJ, 1991.

[7] C. R. Fuller and A. H. von Flotow, "Active Control of Sound and Vibration," IEEE Control Systems, 9-19, December 1995.

[8] S. J. Elliott, P. A. Nelson, "The active control of sound," Electronics & Communication Engineering Journal, 127-137, August 1990.

[9] S. J. Elliott, I. M. Stothers and P. A. Nelson, "A Multiple Error LMS Algorithm and Its Application to the Active Control of Sound and Vibration," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-35, No. 10, 1423-1434, October 1987.

[10] S. J. Elliott, C. C. Boucher and P. A. Nelson, "The Behavior of a Multiple Channel Active Control System," IEEE Transactions on Signal Processing, Vol. 40, NO.5, May, 1992.

[11] D. M. Melton and R. A. Greiner, "Adaptive Feedforward Multiple Input, Multiple Output Active Noise Control System", Proc. IEEE ICASSP, 1992.

[12] D.Guicking and M. Bronzel, "Multichannel Broadband Active Noise Control in Small Enclosure," Proc. Inter-Noise 90, 1255-1258, 1990.

[13] T. J. Sutton, S. J. Elliott, A. M. McDonald, and T. J. Saunders, "Active Control of road noise inside vehicle," Noise Control Eng. Journal 42(4), 137-147, Jul-Aug, 1994.

[14] W. B. Mikhael, P. D. Hill, "Acoustic noise cancellation in a multiple noise source environment," IEEE International Symposium on Circuits and Systems Proceedings, p. 3 vol. 2915, 2399-402, 1988.

[15] R. B. Wallace, R. A. Goubran, "Parallel adaptive filter structures for acoustic noise cancellation," IEEE International Symposium on Circuits and Systems, P.6 vol. 3028, 525-8 vol.2, 1992.

[16] J. Y. Chung, M. J. Crocker, "Measurement of Frequency Response and Multiple Coherence Function of the Noise Generation System of a Diesel Engine," J. Acoust. Soc. Am. 58, 635-642 1975.

[17] M. E. Wang and Malcolm J. Crocker, "On the Application of Coherence Techniques for Source Identification in a Multiple Noise Source Environment," Journal of Acoust. Soc. Am. 74(3), 861-872, September 1983.

[18] Craig M. Heatwole and Robert J. Bernhard, "Reference Transducer Selection for Active Control of Structure-borne Road Noise in Automobile Interiors", Noise Control Eng. 44 (1), 1996 Jan-Feb.

[19] Masato ABE, Guo-Yue Chen and Toshio Son, "A method to increase the convergence speed by using uncorrelators in the active control of multiple noise sources," Proc. Inter-Noise 93, 759-762, 1993.

[20] C. R. Fuller, S. J. Elliott and P. A. Nelson, Active Control of Vibration, Academic Press Inc., London, Great Britain, 1996.

[21] C. R. Fuller, "Active control of sound transmission/radiation from elastic plates by vibrational inputs. I. Analysis," Journal of Sound and Vibration, 136, 1-15, 1990.

[22] R. A. Burdisso, J. S. Vipperman, and C. R. Fuller, "Causality analysis of feedforward-controlled systems with broadband inputs," J. Acoust. Soc. Am. Vol. 94, No. 1, 234-242, July 1993.

[23] B. Widrow and S. D. Stearns, Adaptive Signal Processing, Englewood Cliffs, NJ: Prentice-Hall, 1985.

[24] S. Thomas Alexander, Adaptive Signal Processing, Springer-Verlag New York Inc. New York, NY, 1986.

[25] J. M. Cioffi and T. Kailath, "Fast, recursive-least-squares transversal filters for adaptive filtering," IEEE Trans. Acoust. Speech, Signal Process., vol. ASSP-32, pp. 304-337.

[26] G.A. Clark, S. K. Mitra, and S. R. Parker, "Block implementation of adaptive digital filters," IEEE Trans. Acoust. Speech, Signal Processing, vol. ASSP-29, no. 3, pp. 744-752, June 1981.

[27] J. J. Shynk, "Frequency-domain and multirate adaptive filters," IEEE Signal Process, Mag., vol. 9, no.1, pp. 14-37, Jan. 1992.

[28] Q.Shen and A.S.Spanias, "Time and Frequency Domain X Block LMS algorithms for Single Channel Active Noise Control", Proc. 2nd Int. Congress on Recent Developments in Air- & Structure- Borne Sound and Vibration, PP. 353-360, Auburn, Al, March 1992.

[29] A. V. Oppenheim and R. W. Schafer, Digital Signal Processing. Prentice Hall, Englewood Cliffs, NJ, 1975.

[30] J. S. Bendat and A. G. Piersol, Random Data-Analysis and Measurement Procedure, Second Edition, John Wiley & Sons, Inc., New York, NY, 1986.

[31] G. H. Golub and C. F. VanLoan, Matrix Computations, Second Edition, The Johns Hopkins University Press, Baltimore, MD, 1989.

[32] David S. Watkins, Fundamentals of Matrix Computations John Wiley & Sons, Inc., New York, NY, 1991.

[33] E. Weinstein, M. Feder, and A. V. Oppenheim, "Multi-chjannel signal separation by decorrelation," *IEEE Trans. Speech Audio Processing,* vol. 1, 405-413, Oct. 1993.

[34] S Gerven, D. Compernolle, "Signal Separation by Symmetric Adaptive Decorrelation: Stability, Convergence, and Uniqueness," IEEE Trans. On Signal Processing, vol. 43, No.7, pp. 1602-1612, July 1995.

[35] P. A. Nelson and S. J. Elliot, Active Control of Sound. Academic Press Inc., San Diego, CA92101, 1992.

[36] B. N. Parlett, The Symmetric Eigenvalue Problem, Prentice-Hall, Englewood Cliffs, NJ, 1980.

[37] TMS320C3X User's Guide, Texas Instruments, Inc., Dallas, Tex., 1991.

[38] TMS320C3X Floating-Point DSP Optimizing C Compiler User's Guide, Texas Instruments, Inc., Dallas, Tex., 1991.

[39] TMS320C3X Floating-Point DSP Assembly Language Tools User's Guide, Texas Instruments, Inc., Dallas, Tex., 1991.

[40] James P. Carneal, "Active Structural Acoustic Control of Double Panel Systems including Hierarchical Control Approaches," VPI&SU Thesis, 1992.

[41] A. A. Giordano and F. M. Hsu, Least Square estimation with application to digital signal processing, Wiley, New York, 1985.

[42] C. C. Boucher, S. J. Elliott, and P. A. Nelson, "Effect of errors in the plant model on the performance of algorithms for adaptive feedforward control," IEEE Proc. Radar Signal Process., vol. 138, no.4, pp. 313-319, Aug. 1991.

[43] B. Widrow and E. Walach, Adaptive Inverse Control, Prentice Hall, Englewood Cliffs, NJ, 1996.

[44] D. E. Newland, An Introduction to Random Vibrations, Spectrum and Wavelet Analysis, 3rd ed.1993, New York: Wiley.

Appendix A. TMS320C30 Assembly Functions for the LMS Algorithm

Some special features like circular buffer, parallel addressing can not be fully exploited using C language. In order to take the most advantage of TMS320C30 DSP, Assembly functions callable by C are provided in this appendix.

As mentioned in section 4.3, an adaptive filtering process is basically comprised of two steps:

$$y(k) = \mathbf{x}(k) * w(k) \tag{A.1}$$

$$\mathbf{w}_m(k+1) = \mathbf{w}_m(k) - \mu e(k) x(k-m) \tag{A.2}$$

The first step shown in equation (A.1) represents a linear convolution, and the second step shown in equation (A.2) represents weight vector updating for the LMS algorithm. The two steps can be efficiently and flexibly implemented using three functions. The input vector $\mathbf{X}$ is stored in a circular buffer. The first function named cshiftv.asm is dedicated to put the newest input data into circular buffer to replacing the oldest input data. The second function named conv.asm performs the linear convolution represented by equation (A.1). The third function named updatewt.asm updates the weight vector using the LMS algorithm.

For the single channel LMS algorithm, we can combine the three functions into a more efficient function. However, if we want to implement the multiple reference multiple channel LMS algorithm, the effort to implement a single Assembly function is almost formidable, and the resulting code is not only prone to error, but also hard to modify. With the help of the foregoing three functions, we can implement the multiple reference multiple channel LMS algorithm with great flexibility.

```
****************************************************************************
*          Program: cshiftv.asm      DATE: 03-10-96                         *
*          Version: 2.0                                                     *
*          Author:  Yifeng Tu                                               *
*          Dept of Mechanical Engineering                                   *
*          Virginia Tech.                                                   *
*          Blacksburg, VA24060                                              *
*          description:                                                     *
*                before shift                 after shift                   *
*                     .                            .                        *
*                     .                            .                        *
*          pnew-> x(k)     newest value          x(k-1)                     *
*                x(k-N+1)  oldest value    pnew-> x(k)     newest value     *
*                x(k-N+2)                        x(k-N+1) oldest value       *
*                     .                            .                        *
*                     .                            .                        *
*                     .                            .                        *
*                     .                            .                        *
* This routine stores the input data into circular buffer, the oldest data being replaced by *
* the newest data.                                                          *
*          vlen:    circular buffer length                                  *
*          inp:     newest input data                                       *
*          pxnew:   pointer to the newest value in the circular buffer      *
****************************************************************************
        .version    30
fp      .set        ar3
        .globl      _cshiftv

;>>>>   float *cshiftv(int vlen, float inp, float *pxnew)

_cshiftv:
        push        fp*
        ldi         sp,fp

        ldi         *-fp(2), bk
        ldi         *-fp(4), ar0
        ldf         *-fp(3), r1
        ldf         *ar0++(1)%, r0
        stf         r1, *ar0;
        ldi         ar0, r0              ;return the value

epio_1:
        pop    fp
        rets
        .end
```

```
* ***************************************************************
*          Program: conv.asm     DATE: 03-11-96                  *
*          Version: 2.0                                          *
*          Author:  Yifeng Tu                                    *
*          Dept of Mechanical Engineering                        *
*          Virginia Tech.                                        *
*          Blacksburg, VA24060                                   *
*          description:                                          *
*                              circular  addressing              *
*                                                                *
*             ar0->  h(0)                 .                      *
*           h(1)        ar1->   x(k)         newest value        *
*               .                   x(k-N+1)   oldest value      *
*               .                   x(k-N+2)                     *
*               .                      .                         *
*               .                      .                         *
*             h(N-1)                   .                         *
*                                                                *
*  This routine functions as FIR filter with length N,  pwt points to the beginning  *
*  of vector W,  px points to the newest value of vector X.      *
* ***************************************************************
        .version   30
fp      .set      ar3
        .globl     _conv

;>>>> float conv(int n, float *pwt, float *px)
_conv:
        push    fp
        ldi     sp,fp

        ldi     *-fp(2),bk
        ldi     *-fp(3),ar0  ; ar0 the lowest address of coefficients, i.e. h(0)
        ldi     *-fp(4),ar1  ; ar1 the newest value of input x
        ldi     *-fp(2),rc
        subi    1,rc

        ldf     0.0,r0
        ldf     0.0,r2

        rpts      rc
        mpyf3   *ar0++,*ar1--%,r0
||      addf3    r0,r2,r2
        addf3    r0,r2,r0

epio_1:
        pop     fp
        rets
        .end
```

```
* **************************************************************
*          Program: updatewt.asm      DATE: 03-13-96                *
*          Version: 2.0                                             *
*          Author:  Yifeng Tu                                       *
*          Dept of Mechanical Engineering                           *
*          Virginia Tech.                                           *
*          Blacksburg, VA24060                                      *
*          Description:                                             *
*                                                                   *
*  This routine updates FIR wights using LMS algorithm, pwt points to the  *
*  beginning of vector W,  px points to the newest value of vector X,      *
*                 W(n+1,i) = W(n,i) + u * e * X(n-i);               *
**************************************************************

        .version   30
fp      .set       ar3
        .globl     _updatewt

;>>>> float *updatewt(int n, float *pwt, float *px, float error, float mu)

_updatewt:
        push   fp
        ldi    sp,fp

        ldi    *-fp(2),bk
        ldi    *-fp(3),ar0  ; ar0 the lowest address of coefficients i.e.h(0)
        ldi    *-fp(4),ar1  ; ar1 the newest value of input x
        ldf    *-fp(5),r2   ; error signal
        ldf    *-fp(6),r3   ; mu
        ldi    *-fp(2),rc
        subi   1,rc          ; to make it repeat n times,
                             ; set rc = n-
        mpyf   r3,r2         ; r2 = mu * error
        ldf    0.0,r3
        ldf    0.0,r0

        mpyf3  *ar1--%,r2,r0
        rptb   E_loop
        mpyf3  *ar1--%,r2,r0
||      addf3  r0,*ar0,r3
E_loop: stf    r3,*ar0++

        ldi    *-fp(3),r0

epio_1:
        pop    fp
        rets
        .end
```

Vita

Yifeng Tu was born on February 11, 1971, in a small tranquil town in southern part of China, where he have lived for 17 years until he graduated from Shangrao No. 2 High Scholl. In 1988, Yifeng Tu went to Beijing to pursue his Bachelor of Science in Automotive Engineering from Tsinghua University. After completion of his Bachelor in 1993, he started a career as a software engineer. The position gave him extensive exposure to software engineering, and also great joy of travelling around northern China.

In 1994, Yifeng Tu came to Blacksburg, a peaceful town of southwest of Virginia, to start his graduate program at Virginia Tech., where he joined the Vibration and Acoustics Labs of Mechanical Engineering Department. The three years' stay in Blacksburg has brought upon Yifeng terrific life style along with excellent education he will enjoy for lifetime. He is currently finishing his Master of Science and working on an exciting project with other VAL members.

Yifeng Tu