isvr Institute of Sound and Vibration Research

Multichannel Algorithms for Active Sound Profiling

Lewis E. Rees

ISVR Technical Memorandum No 958

November 2005

University of Southampton

SCIENTIFIC PUBLICATIONS BY THE ISVR

Technical Reports are published to promote timely dissemination of research results by ISVR personnel. This medium permits more detailed presentation than is usually acceptable for scientific journals. Responsibility for both the content and any opinions expressed rests entirely with the author(s).

Technical Memoranda are produced to enable the early or preliminary release of information by ISVR personnel where such release is deemed to the appropriate. Information contained in these memoranda may be incomplete, or form part of a continuing programme; this should be borne in mind when using or quoting from these documents.

Contract Reports are produced to record the results of scientific work carried out for sponsors, under contract. The ISVR treats these reports as confidential to sponsors and does not make them available for general circulation. Individual sponsors may, however, authorize subsequent release of the material.

UNIVERSITY OF SOUTHAMPTON

INSTITUTE OF SOUND AND VIBRATION RESEARCH

SIGNAL PROCESSING & CONTROL GROUP

Multichannel Algorithms for Active Sound Profiling

by

Lewis E. Rees

ISVR Technical Memorandum N° 958

November 2005

Authorised for issue by
Prof R Allen
Group Chairman

Acknowledgements

The author would like to thank InMAR both for sponsorship and support, specifically all those partners involved in this project, including: VTT - Finland, Ford - Germany, Techno First - France and Rieter - Switzerland.

Contents

List of Figures

Introduction

In the design of an active sound profiling system it is important that all the components of the system set-up are chosen carefully such that the system is as robust and efficient as possible. This is particularly true for multichannel control systems, in which multiple loudspeakers are used to control the sound at multiple error microphones, which is the subject of this report and follows on from earlier work on single channel control systems [1]. Of particular concern to the implementation of such a control system are the trade-offs involved in achieving specific targets. For example, the degree of control that is achieved versus the amount of power required to do so, and whether or not individual signals at individual microphones are to be achieved or whether a global average within the cabin is sufficient. The choice of strategy can vary both power to which the control system has to use, this is referred to as control effort, and the robustness of the system; how easily the system can become unstable. To implement these different strategies, adaptive algorithms are required to alter the control signal which in turn will drive the loudspeakers within the cabin. Such algorithms have previously been used in multichannel systems whose aim is to cancel sound within the cabin, but not to reshape the soundfield as required in sound profiling. It is the future aim of the work reported here to produce a working multichannel car system that can achieve cancellation, attenuation and enhancement of a given engine order spectrum within the car cabin, such that a predetermined "target" spectrum is produced within the car. At this point in time it is unclear to what degree of control is required to achieve the subjective goals within a vehicle cabin, hence this report presents several algorithms that can be used depending on the final control strategy chosen.

This document has been written to be both a general report and a specialised technical paper that highlights specific areas that are relevant to the InMAR work area 4.5. The report first explains the multichannel problem in the car cabin, then the functioning of the algorithms is covered, followed by a performance comparison of the algorithms in terms of stability and control effort, including a discussion to conclude which algorithms are most suitable for a vehicle's control system.

The Multichannel Problem

In the current vehicle prototype control system there are only 2 error microphones and 2 control loudspeakers to perform active sound profiling at the drivers head position, however, it is the aim of the project to create a system that can control the engine sound pressure level (SPL) over the entire soundfield within the cabin, such that all passengers experience the altered soundfield and are thus subjected to an increase in sound quality. To create such a system, many error microphones and control loudspeakers, may be required, not necessarily in equal quantities, as depicted in Fig. 1. Ideally the error microphones will be placed close to the location of the driver's and passenger's heads, to give an approximate measure of the pressure level experienced at the ear. Thus, it can be assumed by basing the standard vehicle passenger capacity at four, including the driver, that for a complete multichannel system in the order of 8 error microphones and 8 control loudspeakers will be required. In such a system, containing this many microphones, the sound profiling problem is much more complicated since several different performance criteria can be defined. These performance criteria vary in

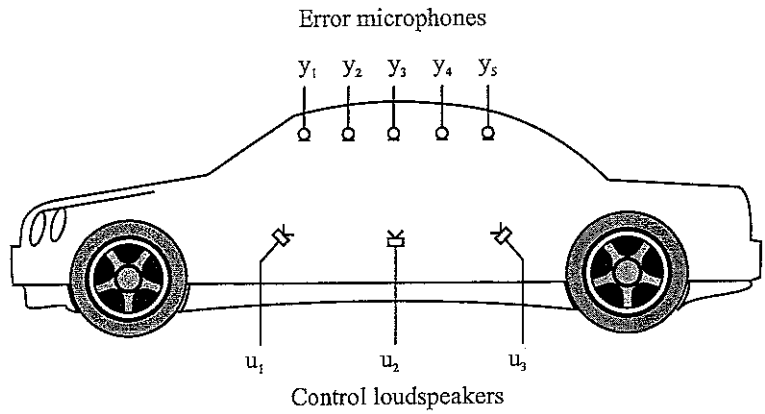


Figure 1: The soundfield within a car cabin can be controlled using an array of error microphones and control loudspeakers.

terms of the properties required by the output signal, and include:

{1} Maintaining the amplitude and phase of each microphone output to be similar to target signals that possess individual amplitude and phase information

{2} Maintaining only the amplitude at each error microphone similar to individual target amplitudes, but with no concern to their relative phases

{3} Maintaining the sum of mean squared outputs to be similar to an overall mean square pressure target, but with no concern for the relative magnitude and phases of the individual output signal.

Each of these strategies will have their relative merits and failings, of which many are unknown at this point, however, it is expected that strategy {1} will be most demanding in terms of stability and control effort, whilst strategy {3} will be the least demanding. In active sound-profiling, rather than active noise control, great attention has to be paid to the properties of the signal produced at the error microphones. In active noise control, it is of little importance what phase or amplitude the residual signal has from frequency to frequency, as the aim of the control system is to merely reduce the SPL to as low a level as possible. In active sound profiling, however, the subjective sound quality is the primary concern, which, in special cases, can be greatly altered by the relative phases and amplitudes between engine orders. The relative phase and magnitudes of specific engine orders can give the car engine the signature sound that is associated with high levels of subjective sound quality. One such example is the 'warble'-like sound that is produced by a typical V8 engine. This warble is generated by the intermodulation of engine orders in the range 1-6, of which the phase and relative amplitudes is inherent to the production of the sound. Thus, to implement a control system that is not concerned with maintaining these relative properties could decrease the sound quality of the engine rather than increase it. This is not always the case as for many engine orders the phase is not important to the interpreted sound quality of the engine. It is suspected that for a key engine order, such as the 2^{nd} engine order in a 4-cylinder vehicle, the phase of the order is not significant to the subjective sound quality, and thus control strategies that are not concerned with the phase of the signal at the error microphone may be suitable for such a

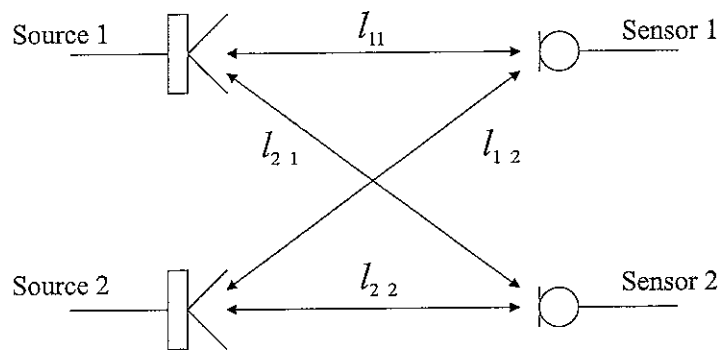


Figure 2: Schematic showing the secondary acoustic paths of the plant between sources and sensors.

control system. It should, however, not be forgotten that such psychoacoustic effects do exist and can be altered by active sound profiling.

In multichannel systems, such as that proposed for the vehicle here, the interaction between the error microphones and control loudspeakers plays a very important part in the boundaries of the system and how easily control can be achieved. Each microphone and loudspeaker in the control system, and to a greater extend the amplifiers and digital-to-analogue conversion hardware, has an associated amplitude and phase frequency-response. This is primarily dictated by the relative positions of the microphones and loudspeakers and the acoustic response of the cabin enclosure, and is often referred to as the 'plant' of the system. This series of frequency responses that make up the plant can be represented by a set of complex equations, which can in turn be simplified into matrix form. For a simple two-channel system under anechoic conditions, with 2 error microphones and 2 control loudspeakers as shown in Fig. 2, the complex plant matrix for a given frequency, only taking into account the acoustic path length such that loudspeaker and microphone responses are considered to be flat, can be given by

$$\mathbf{G}(j\omega) = \begin{bmatrix} \frac{A}{l_{11}} e^{-jkl_{11}} & \frac{A}{l_{12}} e^{-jkl_{12}} \\ \frac{A}{l_{21}} e^{-jkl_{21}} & \frac{A}{l_{22}} e^{-jkl_{22}} \end{bmatrix} \tag{1}$$

where the distances l_{mn} correspond to those in Fig. 2, m and n are the number of sensors (microphones) and sources (loudspeakers), respectively. In a simple model, such as this, the phase responses of the plant represent the acoustic delays between control loudspeakers and error microphones, which is the time that the acoustic waves take to propagate through air at the speed of sound. This matrix can now be used to calculate the contribution of each control signal (the output from the adaptive filter) at the error microphone locations, such that if we assume there is no disturbance signal present

$$\mathbf{y} = \mathbf{G}\mathbf{u} \tag{2}$$

where $\mathbf{y}$ is a column vector of complex outputs measured at the error microphones and $\mathbf{u}$ is a column vector of complex control signals sent to the control loudspeakers, all at a single frequency. It is clear from (1) and (2) that the location of the microphones and loudspeakers can greatly affect the control signals required to produce a given output at the error microphones.

This effect of the plant can be reduced substantially by choosing a control strategy that will achieve its given criterion with a minimal amount of control effort and maintainable stability.

It is uncertain which strategy, {1}, {2} or {3}, is the most suitable for a vehicle sound profiling problem, until it is it is completely clear what the criteria of the control system are. If, for example, it has been chosen that it is desirable that the driver hears a different engine spectrum to that of the passengers, for the advantage of driver feedback and passenger comfort, then the target spectra at the drivers microphones will be different to those of the passengers. To implement such a control strategy one of performance criteria {1} or {2} will need to be chosen. However, if it is chosen that both driver and passengers are to hear the same engine spectra then, depending on the properties of the plant, it may be adequate to use performance criteria {3}, which may benefit the system in terms of required control effort and system stability. Albeit, each of these different control strategies will require a corresponding adaptive algorithm to ensure the target criteria is met in each of the cases.

The Algorithms

At the heart of an adaptive feedforward control system is the algorithm that updates the weights of the adaptive filter. This algorithm must be tailored both to produce the best results for the initial system requirements, but also be able to be adaptive to suit any new changes that may arise in the development of the control system and will normally incorporate an internal model of the plant response. One aspect that can be particularly demanding for an active sound profiling algorithm is the ability to remain stable when plant model errors are present. The nature of an acoustical system such as the interior of an automobile means that the physical plant is continuously changing. These changes may be caused by variations such as temperature, humidity, or alterations to the acoustic enclosure response, caused by movement of the driver/passenger or the opening of a window or sun-roof. This dynamic nature of the physical plant means that the fixed plant model will never be a perfect estimate, and in addition is bound by the resolution of the digital filter used to model the plant. To make matters more complicated it is desirable for the algorithms to use a minimal amount of control effort to achieve the target spectrum. The use of excessive control effort to produce the target spectrum can put unnecessary restrictions on the equipment used for sound generation, such as the amplifiers and loudspeakers. The size of the loudspeakers used in a car is highly restricted and using a minimal control effort algorithm allows loudspeakers of lower power and smaller size to be adequate to achieve control. Algorithms currently exist that perform well either in terms of stability to plant model errors [1] or control effort [2], but not to both especially at high enhancement gains.

The following sections contain brief descriptions of the functioning of the four algorithms compared in this report: the command-FXLMS, phase scheduled command-FXLMS (PSC-FXLMS), automatic phase command-FXLMS (APC-FXLMS) and the magnitude-controlled FXLMS without and with the sum of squared outputs control strategy (SOSO-FXLMS). For a more in-depth working of the algorithms the reader is referred to *Adaptive Algorithms for Active Sound-Profiling*, 2005 by Rees and Elliott [1] for the command-FXLMS, PSC-FXLMS and APC-FXLMS, and appendix A on the magnitude-controlled FXLMS and sum of squared

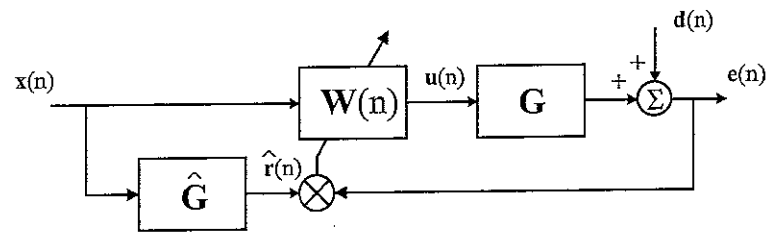


Figure 3: Block diagram of the FXLMS adaptive feedforward control algorithm.

output strategy.

Frequency-Domain Filter-Weight Adaption

All the algorithms used in this investigation are frequency domain based algorithms. This means that all the signals required for the update of the adaptive filter are first transformed from the time-domain into the frequency-domain via an FFT (fast Fourier transform), where the magnitude and phase of each frequency to be controlled is represented as a complex number. The FFTs used in the algorithms buffer n samples of the signal that equate to one period of the fundamental reference frequency and output a new vector of complex values every n samples. Therefore, update of the adaptive filter is not instantaneous, and although under some circumstances this could be detrimental to the convergence speed of the system it is thought to make no noticeable difference in this application. This method has been used as the new algorithms, require magnitude information of the error signals. This can be achieved by computing a moving time-average and updating the weights every sample, but it is more computationally efficient to do all computation in the frequency domain. To aid in the analysis of all algorithms, the command-FXLMS has also been implemented in the frequency-domain such that direct comparisons with the other algorithms can be made on a level playing field. Thus, all variables referred to in the following sections are complex.

Command-FXLMS

The algorithm currently implemented in the prototype control unit of the vehicle is the command-FXLMS. This algorithm is based on the filtered-reference least mean square (FXLMS) algorithm widely used in feedforward active noise control. The FXLMS, shown as a block diagram in Fig. 3, is a simple algorithm that uses an instantaneous error output to predict the gradient of the error surface of a cost function, $J = \mathbf{e}^{\mathrm{H}}\mathbf{e}$, where

$$\mathbf{e}(n) = \mathbf{d}(n) + \mathbf{G}\mathbf{u}(n) \tag{3}$$

where $\mathbf{e}(n)$ is a column vector of complex output error signals at the n^{th} iteration of the algorithm, $\mathbf{d}(n)$ is a column vector of complex disturbance signals, $\mathbf{u}(n)$ is a column vector of complex control signals and $\mathbf{G}$ is a matrix of complex values that describes the plant. It is the aim of the algorithm to vary the value of $\mathbf{u}(n)$ by the adaption of a matrix of complex coefficients, $\mathbf{W}$, such that the cost function is minimised. The gradient term is then used, via

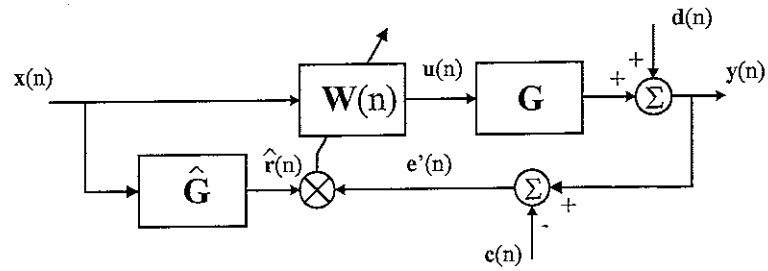


Figure 4: Block diagram of the command-FXLMS.

the multiplication of a convergence coefficient, α, that regulates the size of the gradient term, to adapt the control signal, such that

$$\mathbf{u}(n+1) = \mathbf{u}(n) - \alpha \hat{\mathbf{G}}^{\mathrm{H}} \mathbf{e}(n) \tag{4}$$

where $\mathbf{u}(n+1)$ is the updated complex control signal vector and $\hat{\mathbf{G}}$ is the plant model matrix. The plant model is a digital filter that is used to emulate the effects of the physical plant, $\mathbf{G}$, such that the algorithm converges well towards the optimum value of $\mathbf{u}(n)$ and remains stable in doing so.

The FXLMS, however, only possesses the ability to cancel signals, i.e. to reduce the output towards zero. For active sound profiling the algorithm needs to be capable of producing a residual error signal, in which the output may be less than, equal to or greater than the disturbance signal, hence the FXLMS algorithm is not sufficient to fulfil this task. Although, with a slight alteration the FXLMS can be fooled into converging to a non-zero value. This is done by replacing the error, $\mathbf{e}(n)$, in the gradient term to update the adaptive filter weights, with a pseudo-error which is equal to

$$\mathbf{e}'(n) = \mathbf{e}(n) - \mathbf{c}(n) \tag{5}$$

where $\mathbf{c}(n)$ is a column vector of complex target or command signals. Thus, by using a new control signal update equation, given by

$$\mathbf{u}(n+1) = \mathbf{u}(n) - \alpha \hat{\mathbf{G}}^{\mathrm{H}} \mathbf{e}'(n) \tag{6}$$

the output now converges towards the target vector, rather than to zero. This alternate form of the FXLMS is known as the 'command-FXLMS', and is shown as a block diagram in Fig. 4. In the diagram, $\mathbf{e}(n)$ has been replaced with $\mathbf{y}(n)$ to signify that the value is an output and not an error.

The command-FXLMS is known to have similar properties to the FXLMS [1], but can also use more effort than necessary. The command signal, $\mathbf{c}(n)$, can have an arbitrary phase as well as amplitude, which if out of phase with the disturbance signal, $\mathbf{d}(n)$ requires the control signal, and thus the control effort, to increase unnecessarily in comparison to if $\mathbf{c}(n)$ and $\mathbf{d}(n)$ were in phase. This would put small control loudspeakers under unnecessary strain to deliver low frequencies at high outputs, which, in extreme cases, could lead to instability. One way of reducing the amount of control effort used would be to measure the primary path phase response, i.e. the time delay from the reference signal generation to the error microphone. This transfer function could then be used to filter the command signal, and assuming no

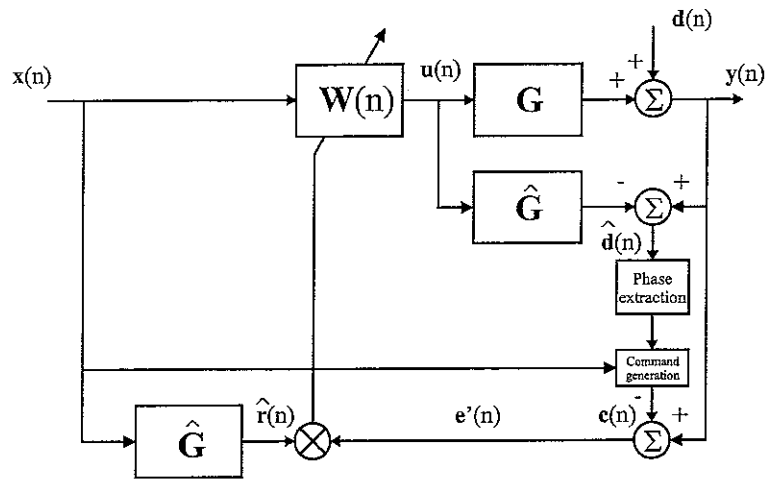


Figure 5: Block diagram of the PSC-FXLMS.

other variations occur, the disturbance and the filtered command signals should be in phase and thus use a minimal amount of control effort to achieve the target output. However, as previously discussed, it is unlikely that the primary path will remain unchanged and will be the same for individual cars. For this reason it would be more suitable to use an algorithm that can predict the disturbance signal phase while online. This method is used by the PSC- and APC-FXLMS algorithms that have the ability to align the phase of the command signal to that of the disturbance signal, but do so to different degrees. In addition, although not critical to the engine spectrum, if the disturbance and command signals are kept in phase this would eliminate any cross-order phase issues that may arise of a subjective nature.

Phase Scheduled Command-FXLMS (PSC-FXLMS)

There exist several active sound profiling algorithms that attempt to ensure that the command signal is in phase with the disturbance and thus reduce the amount of control effort required to achieve the target output [1,2]. Unfortunately, many of these algorithms suffer from instability problems when the magnitude of the plant model is not particularly accurate, especially at high enhancement system gains, i.e. when the ratio of command to disturbance magnitudes is large. However, an algorithm does exist that maintains both stability in the face of plant model mis-estimation and achieves accurate target signal convergence at low control effort. This algorithm is known as the phase scheduled command-FXLMS (PSC-FXLMS). The PSC-FXLMS algorithm, schedules the phase of the command signal to that of the disturbance signal's by calculating an estimate of the disturbance, $\hat{\mathbf{d}}(n)$. This estimate of the disturbance signal is achieved by sending the control signal through an internal model of the plant and subtracting the system output, $\mathbf{y}(n)$, as shown in Fig. 5. The phase of the estimate is then calculated and the command signal generated with the same phase but with a new command magnitude. By extracting the phase and re-generating the command signal, rather than simply passing $\hat{\mathbf{d}}(n)$ through a gain, instabilities due to plant model magnitude errors are eliminated. The algorithm is still prone to errors in the plant model phase, however, and due to the use of the plant model once again being used in the internal model to predict the disturbance signal the PSC-FXLMS has less stability than the command-FXLMS.

Automatic Phase Command-FXLMS (APC-FXLMS)

To increase the stability of the PSC-FXLMS to plant model phase errors at high system gains, as often required in active sound profiling, the degree of scheduling of the command signal phase can be altered. The phase of the command signal is changed such that when the system gain is greater than unity the phase is only scheduled to a fraction of the disturbance signal estimate's, such that

$$\phi_c = \frac{2\hat{D}}{C+\hat{D}}\phi_{\hat{d}} \tag{7}$$

where C and ϕ_c are the command signal's magnitude and phase, respectively, and $\hat{D}$ and ϕ_d are the disturbance estimate signal's magnitude and phase, respectively, for a given channel. This new algorithm is called the automatic phase command-FXLMS (APC-FXLMS) [1], and increases the stability of the algorithm to plant model phase errors, but at the cost of increased control effort relative to the PSC-FXLMS when the plant model is accurate. However, the APC-FXLMS has the ability to be 'tuned', such that the control effort savings are increased. This adaption of the APC-FXLMS is known as the 'optimised APC-FXLMS' and is discussed further in appendix B. In this report the APC-FXLMS will be only used in its standard form, as above in (7).

Magnitude-Controlled FXLMS with Individual Magnitudes Specified

As discussed earlier, it may not be necessary for the phase of the output signals to be of any specific value, but only that the magnitude of the signal is maintained at a specified level. It would seem that this reduced constraint on the production of the control signals would result in a reduction in control effort, as unlike full active cancellation, sound profiling can be achieved from multiple solutions. One such algorithm that performs in this manner is the magnitude-controlled FXLMS, which adapts the weights of the adaptive filter using a gradient term who's step size is controlled by the difference between the magnitudes of the output signal, $\mathbf{Y}(n)$, and the command signal, $\mathbf{C}$, which is a non-complex scalar matrix, as no phase information is required. This algorithm is unique in the way it uses a positive convergence coefficient, when $\mathbf{Y}(n) > \mathbf{C}$, to adapt the filter weights, which in essence makes the algorithm unstable and increases $\mathbf{u}(n)$. When $\mathbf{Y}(n) < \mathbf{C}$, however, the effective convergence is negative, as in the FXLMS. This instability is controlled, however, by the magnitude difference term just discussed. As derived in appendix A, the control signal update equation for the magnitude-controlled FXLMS is given by

$$\mathbf{u}(n+1) = \left(1 - \left|\rho[\mathbf{C}-\mathbf{Y}]^{-1}\right|\right)\mathbf{u}(n) + \alpha[\mathbf{C}-\mathbf{Y}]\mathbf{Y}^{-1}\hat{\mathbf{G}}^{\,\mathrm{H}}\mathbf{y}(n) \tag{8}$$

where $\mathbf{u}(n+1)$ and $\mathbf{u}(n)$ are the new and previous complex control signal vectors, $\mathbf{C}$ and $\mathbf{Y}$ are diagonal matrices of the command levels and the output levels, respectively, and $\hat{\mathbf{G}}$ is a matrix of complex plant model responses between sources and sensors, as shown in Fig. 2. It should be noticed that in comparison to the command-FXLMS update equation, (6), (A.10) has an additional term that in multiplied by the current control signal, $\mathbf{u}(n)$. This is known as a leakage term and allows the algorithm to keep on converging towards the minimum control effort solution, even when the output has converged to the target values. ρ is a parameter

that defines the degree to which the leakage term contributes to (A.10). This is discussed in further detail in appendices A and C.

Although the magnitude-controlled FXLMS was originally thought to reduce the control effort load compared to previous algorithms, it is shown analytically in appendix A that the optimum value of the control signal that also results in the minimum control signal is equal to that of the command signal set to the same phase as the disturbance signal. Therefore, the magnitude-controlled FXLMS can not out perform the PSC-FXLMS algorithm if the phases of the disturbance were known, even though it has less performance criteria. This, however, does not mean that the magnitude-controlled FXLMS converges upon the optimal solution in the same way as the PSC-FXLMS. As the magnitude-controlled FXLMS arrives at the same optimum value of $\mathbf{u}$ as the PSC-FXLMS, it would be acceptable to assume that the magnitude-controlled FXLMS could be used in place of the PSC-FXLMS and potentially save on the additional computation that is required to align the command signals phase to that of the disturbance signal's. Unfortunately, the magnitude-controlled FXLMS also requires almost as much computation to derive the magnitude of the output signal, and thus no real advantage would be gained. The analytical stability properties of the magnitude-controlled FXLMS are also unknown, but a preliminary simulated analysis is included in this investigation.

Magnitude-Controlled FXLMS for the Sum of the Mean Square Outputs (SOSO-FXLMS)

As discussed in the multichannel section, it may not be necessary for each error microphone to be driven to a specific output level, but instead the overall soundfield of the car interior be driven towards some predetermined average SPL. In this case the individual phase and magnitudes measured at each of the microphones are no longer used to update the adaptive filter, but rather a 'sum of the mean square outputs' from all of the microphones combined. This removes the constraints of phase and magnitude at each error microphone and allows the potential for a lower control effort to be achieved through a sum of any control signal combination. The phase and magnitude of each control signal are still altered progressively like in all the other algorithms, but are allowed to converge on any value as long as the sum of all the squared outputs converges on the mean squared target value. As with the magnitude-controller FXLMS, to ensure that the optimum control vector is achieved, that is to achieve the target output with the minimum amount of control effort, a leakage term is included in the algorithm which is proportional to the control effort. Thus, as described in the appendix A, the update equation for the control signal vector is given by

$$\mathbf{u}(n+1) = \left[1 - \rho \mathbf{u}^{\mathrm{H}}(n)\mathbf{u}(n)\right] \mathbf{u}(n) + \alpha \left[\frac{C - \sqrt{\mathbf{y}^{\mathrm{H}}(n)\mathbf{y}(n)}}{\sqrt{\mathbf{y}^{\mathrm{H}}(n)\mathbf{y}(n)}}\right] \hat{\mathbf{G}}^{\mathrm{H}}\mathbf{y}(n) \tag{9}$$

where C is the mean command magnitude for the entire microphone array and ρ is the control effort weighting parameter. The difference between C and the root sum of squared outputs, $\sqrt{\mathbf{y}^{\mathrm{H}}(n)\mathbf{y}(n)}$ which determines whether the algorithm increases or decreases the control signals, is normalised by $\sqrt{\mathbf{y}^{\mathrm{H}}(n)\mathbf{y}(n)}$ to make the algorithm less sensitive to the magnitude of $\mathbf{y}(n)$. The size of ρ dictates the amount of control effort that is allowed to be used to achieve the target output. If this is too big, then the converged output will not be accurate to that of C, while if it is too small then the algorithm will take a very long time

to achieve the minimum control effort even once the output has converged on the command value.

Algorithm Performance Comparison

This section presents the results of simulations conducted using the five algorithms previously discussed in this report. Although the aim of this work is to investigate the most appropriate multichannel algorithm, analysing the performance of an algorithm in a multichannel system can often be confusing, hence this section will also include results showing the algorithms used in a single-channel system for the benefit of understanding the way in which the algorithms function. It should be noted that the comparisons made will also compare the relative merits of the control strategies chosen.

The procedure of sound profiling has four modes of control: cancellation, attenuation, neutral and enhancement. The most demanding of these four modes both in terms of stability and control effort is enhancement and for this reason the results of the simulations presented are all generated with the algorithms in enhancement mode. Thus, it is assumed that if stability is retained for an algorithm in enhancement mode then stability is also retained in all other modes and will be covered later in the report.

The level of enhancement that the algorithms are expected to perform has been chosen from measured/target values for the vehicle in question, which show that the greatest enhancement required for any order at any point over the complete engine speed and load range is +17 dB, which corresponds to a gain of ≈ 7.1. For the simulations presented in this section the stable converged enhancement gain is set to 8, to account for variations in the measured spectrum that may occur between individual cars.

To test the extent of the algorithms stability in the single channel simulations, errors were deliberately introduced into the plant model in the form of a phaseshift. This phase error is varied in incremental phase steps of $\pi/8 \approx 22°$, which for a sample rate of 16 samples per period equates to one sample. All algorithms are unaffected by plant model errors in magnitude to the same extent as the FXLMS algorithm, and thus magnitude errors are excluded from this investigation.

It is important that the reader understands that for all of the new algorithms designed to fulfil both magnitude and phase criteria the value of the control effort is dependent on both the phase of the disturbance signal and the phase error in the plant model. In the case of the PSC-FXLMS and APC-FXLMS the command signal is generated using a model of the disturbance signal which in turn is estimated using the plant model. Therefore, if the the phase response of the plant model is incorrect neither will be the phase of the command signal, relative to that of the disturbance signal. Depending on whether the disturbance signals phase leads or lags the reference frequencies and whether the plant model's phase response leads or lags the physical plant's means that the phase difference between the disturbance and command signals can vary significantly. This means that under certain conditions some algorithms that would not normally be expected to have a lower control effort value than other algorithms may

do so, which can be seen in Fig.'s 6(c) and 6(d). Therefore, when analysing the control effort performance of an algorithm it is important to know under what conditions the algorithms are most likely to spend the majority of the time. It must be assumed that for the majority of the time the plant model estimate is accurate to that of the physical plant and thus when comments are made on the control effort of the algorithms this is done so under the conditions of a perfect plant model, i.e. $\hat{G}(z)/G(z) = 1$.

A of all the MATLAB programs used in this report can be found in appendix D.

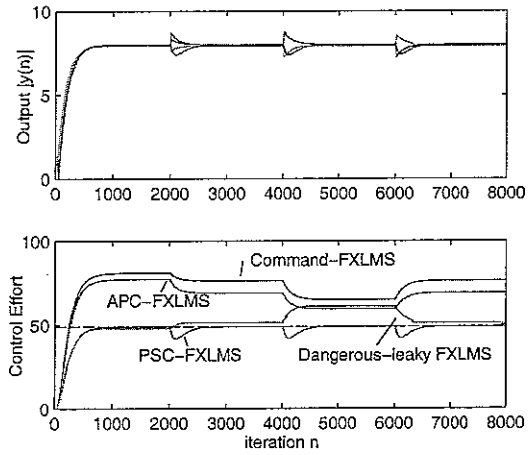
Single Channel Simulations and Results

The results of simulations for all the algorithms, except the SOSO-FXLMS, in a single channel system are presented in Fig. 6. The SOSO-FXLMS has been excluded from the single channel simulations, as in the limit of one channel the SOSO-FXLMS becomes the magnitude-controlled FXLMS. The figure shows five plots of the algorithms, for different plant model phase errors, profiling a pure tone disturbance signal whose phase, relative to the reference signal, changes from $\pi \rightarrow 3/4\ \pi \rightarrow 1/2\ \pi \rightarrow 3/4\ \pi$. These phase transitions can be seen by the slight blips in the convergence of the output magnitude and the variation in the control effort. The reason for the phase shift back to $3/4\ \pi$ is to show that the algorithms command signal is adapting to the disturbance signal and not just cycling through the phase, specifically in the case of the magnitude-controlled FXLMS.

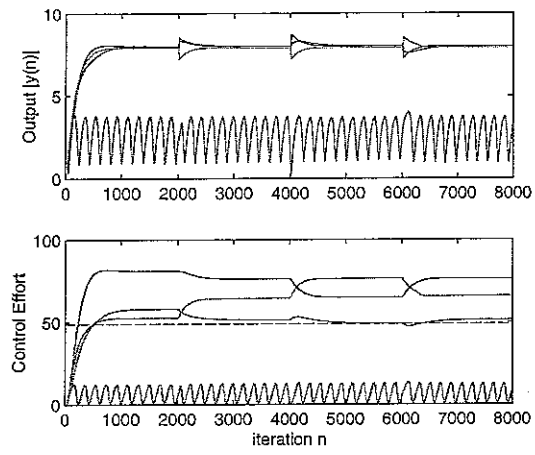
For these simulations the leakage factor, for the magnitude-controlled FXLMS algorithm, has been set to a value of $\rho = 0.0001$. The optimum value of ρ required for implementation in the a particular system is not known, but it is important that the value of ρ is addressed. As discussed in appendix A, if the value of ρ is too high then the algorithm does not converge accurately enough to the desired output level. On the other hand, if the value of ρ is too low, the speed of the algorithm to adapt to find the lowest control effort, once convergence has occurred, is slowed.

From Fig. 6 it can be seen that, when the algorithms are stable, convergence upon the target output is achieved through all disturbance signal phaseshifts. The most noticeable characteristic of the plots in Fig. 6 is that the PSC-FXLMS (green line) enters a limit cycle for all values of $\hat{G}(z) \neq G(z)$. However, when the algorithm is stable it performs best out of all of the algorithms, in terms of control effort. As previously discussed the control effort used by each algorithm varies according to the phase of the disturbance signal and the value of $\hat{G}(z)$, but on average the command-FXLMS uses the greatest amount of control effort, while due to the small leakage factor the magnitude-controlled FXLMS and the APC-FXLMS perform similarly and both have lower control effort than the command-FXLMS, when $\hat{G}(z) = G(z)$. As shown in appendix A, the minimum control effort achievable by the magnitude-controlled FXLMS in theory is the same as that of the PSC-FXLMS, but due to the slow convergence towards the minimum control effort, the magnitude-controlled FXLMS sometimes uses a larger control effort than the PSC-FXLMS.

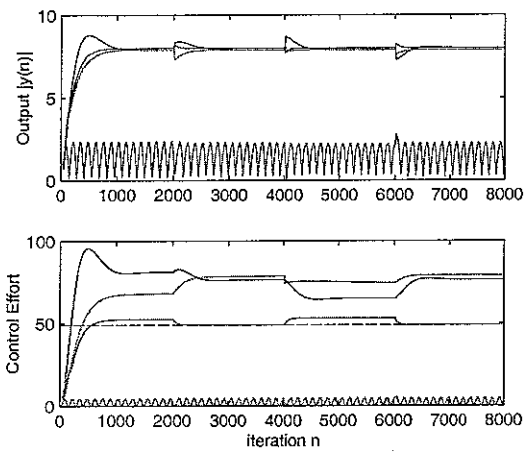
In cancellation mode, not shown here, all algorithms behave very similarly and so the results are not shown. There is only one combination of magnitude and phase that can achieve



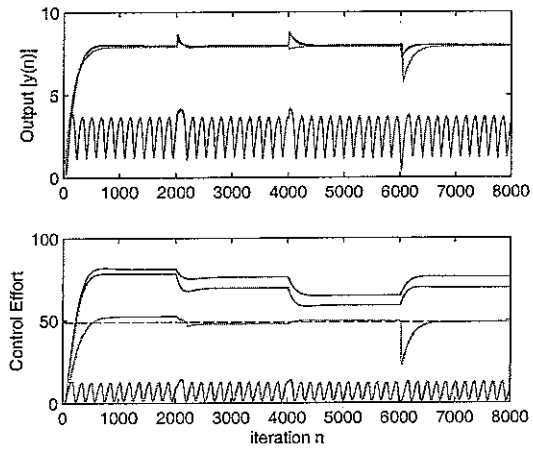
(a) $\hat{G}(z)/G(z) = 1$



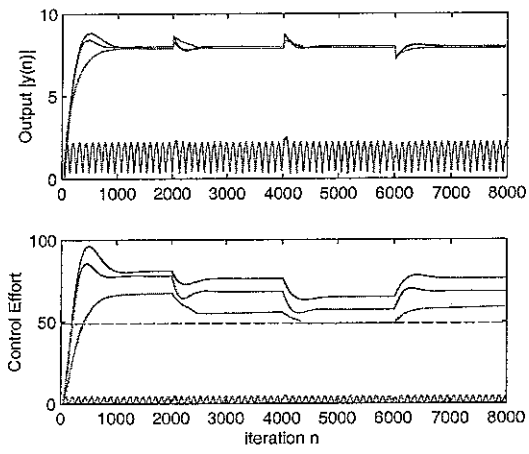
(b) $\hat{G}(z)/G(z) = z^{-1}$



(c) $\hat{G}(z)/G(z) = z^{-2}$



(d) $\hat{G}(z)/G(z) = z^{+1}$



(e) $\hat{G}(z)/G(z) = z^{+2}$

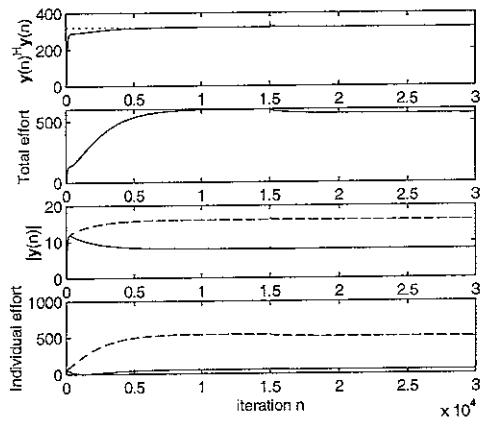
Figure 6: Convergence for algorithms in single channel enhancement mode: command-FXLMS (blue), PSC-FXLMS (green), APC-FXLMS (red) and magnitude-controlled FXLMS (cyan). $|d(n)| = 1$, $C = 8$, $\alpha = 0.1$, $\rho = 0.0001$, $f_n = 1/16$, $N = 16$ and $\hat{G}(z) = 1$. The dashed line shows the minimum control effort achievable.

complete cancellation of the disturbance signal in this mode, and thus all the algorithms use the same amount of control effort to achieve this. Inaccuracy of the plant model does not affect the amount of control effort used, but does affect the rate of convergence, output accuracy and the stability of the system. In attenuation and neutral modes, however, the algorithms once again behave differently. The command-FXLMS uses the greatest control effort, while the PSC- and APC-FXLMS,which are identical in these two modes, use minimum control effort unless plant model inaccuracies are present. The magnitude-controlled FXLMS, although converges quickly on the desired output, takes significantly longer to converge on the minimum control effort. This is because the leakage term has to be small to allow the algorithm to converge accurately, but slows the adaption of the control signal phase. All algorithms in these modes are highly stable.

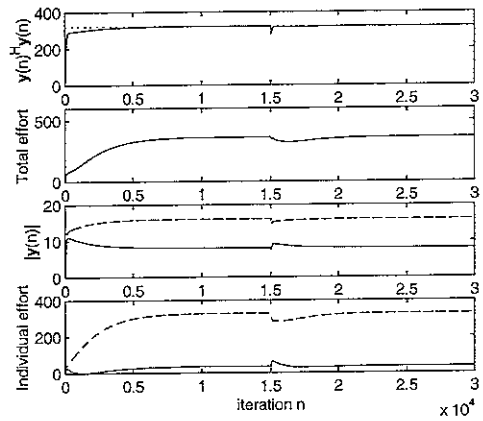
Multichannel Simulations and Results

Fig. 7 shows the convergence of all five algorithms for an equal gain enhancement in a 2 channel (2 loudspeakers and 2 microphones) system, with a plant response described by (1). The disturbance signals have been given slightly different magnitudes, as they would be in a real-world system, and the phase of the disturbance signal changes from $\pi \rightarrow 3/4\ \pi$ half way through the simulation. The simulation has been conducted with a perfect plant estimate, such that $\hat{G}(z) = G(z)$ where $G(z)$ is symmetric. This allows a control effort comparison to be made with no complications of instability or control effort increase due to plant estimate instabilities. From figures 7(a) to 7(d), which all fulfil criteria {1} as discussed previously, the command-FXLMS has the greatest control effort, closely followed by the APC-FXLMS. Also, as seen in the single channel simulations, the PSC-FXLMS and the magnitude-controlled FXLMS have the best control effort, which are approximately equal, but the magnitude-controlled FXLMS takes a considerable amount of time to achieve the minimum control effort, even though the output signals have converged to their respective target signals. It should also be noted that if the APC-FXLMS was optimised, as discussed in appendix B, to the correct system gain and was stable, it could also achieve the minimum control effort. However, even the algorithms that achieve the minimum control effort for criteria {1} require a relatively large amount of control effort to deal with the cross coupling of the plant. In contrast, it can be clearly seen, from Fig. 7(e), that the SOSO-FXLMS uses far less control effort than all other algorithms to achieve the mean square output at both microphones. Only having to fulfil criteria {3}, allows the SOSO-FXLMS use less than a quarter of the control effort required, for example, by the PSC-FXLMS to fulfil criteria {1}. In addition, due to the weak control signal constraint on the algorithm, the convergence of the SOSO-FXLMS to the averaged target output is far quicker than all other algorithms.

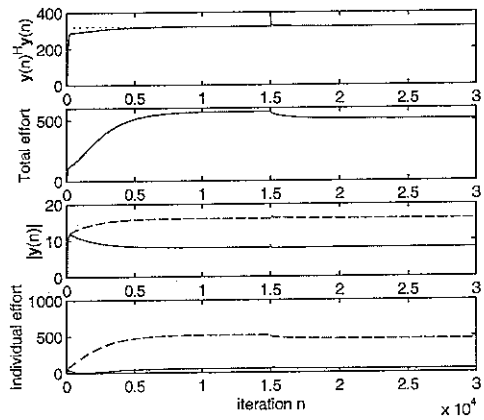
This saving in control effort is largely due to the parameters set in the simulation. If for example, the disturbance signals and target signals were identical for both channels, then the control effort required by the PSC-FXLMS and the magnitude-controlled FXLMS would be the same as that for the SOSO FXLMS. This would, however, require a symmetric plant response, as in the simulation. Even a slightly asymmetric plant response would make it much harder for the algorithms that maintain criteria {1}, on page 2, to achieve their target values and thus increase the control effort.



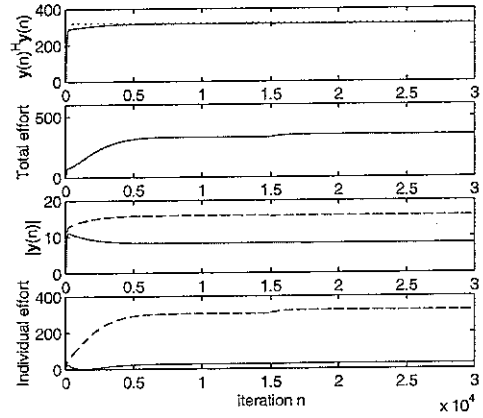
(a) Command-FXLMS



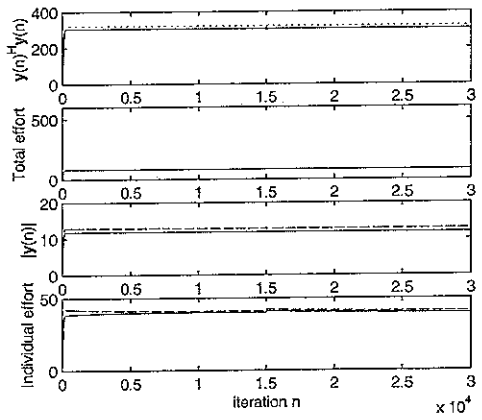
(b) PSC-FXLMS



(c) APC-FXLMS



(d) magnitude-controlled FXLMS



(e) magnitude-controlled-SOSO FXLMS

Figure 7: Convergence for algorithms in enhancement mode for two-channel simulations. $|d(n)| = [1\ 2]$, $C/|d(n)| = 8$, $\alpha = 0.1$, $\rho = 0.0001$, $f_n = 1/16$, $N = 16$ and $\hat{G}(z) = G(z)$, where $G(z)$ is symmetric. Plots containing a single solid line represent the sum of the values for both channels, while those plots with a solid and dashed line show the values for individual channels. The dotted line shows the sum of squared command signals.

Assuming that the system is linear and that adaption of the filter coefficients is slow enough, the evolution of the system to deal with multiple frequency control, would not affect the stability, convergence or control effort of any of the algorithms.

Algorithm Performance Summary

The following section summarises the performance of all five algorithms from the multi-channel simulations conducted and, where relevant, from the single channel simulations.

Command-FXLMS

As can be seen from the plots in Fig.'s 6 and 7 the command-FXLMS has the most predictable behaviour of all the algorithms. This is due to it's simplicity, which results in it's high level of stability and was the primary reason that the command-FXLMS was chosen for the prototype vehicle control unit. As discussed in the introduction, the command-FXLMS uses no information about the phase of the disturbance signal and thus may have to use a large amount of control effort compared to the other algorithms to achieve a desired output. However, as suggested earlier, primary path analysis could be used as additional phase information to filter the command signal and reduce the control effort expenditure. It is unknown whether or not this approach will improve the control effort in practice, but until measurements of the variation of the primary path over time can be taken the effectiveness of this method will remain unknown.

PSC-FXLMS

As discussed in [1], the PSC-FXLMS operates excellently in attenuation mode, but as can be seen from the plots in Fig.'s 6 and 7 does not possess the required stability to fulfil it's primary function at high enhancement gains. At these high gains even small errors in the plant model phase can cause the algorithm to enter a limit cycle, and thus lose control. It is unknown to what accuracy the plant model is required in practice, but it is suspected that the continuously changing environment of a typical car journey will prove to extreme for the algorithm to remain stable at these gains.

APC-FXLMS

It can be seen from Fig.'s 6(a) and 7 that the APC-FXLMS does require less control effort than the command-FXLMS, and at a system gain as high as that required in the extremes of automotive sound profiling the amount of control effort saved over the command-FXLMS is marginally reduced. This reduction in control effort saving brings into question the efficiency of the APC-FXLMS, in terms of the amount of addition computational power that is required to calculate the estimate of the disturbance signal.

However, as discussed in appendix B, the APC-FXLMS can be 'tweaked', such that the control effort savings are maximised. Such tweaking requires an estimate of the system gain, which can be provided from measured and target spectra, so that the APC-FXLMS produces control effort savings as high as the PSC-FXLMS, but with increased stability. Provided that the estimated measured/target gain is reasonably accurate the algorithm will remain stable, but until full real-world implementation of this version of the algorithm can be investigated the stability of the algorithm is not certified, unlike the original version.

Magnitude-Controlled FXLMS

The magnitude-controlled FXLMS is still a relatively unknown algorithm. Unpredictably, as shown in Fig. 6, it seems to be the most stable of all the algorithms, even more so than the command-FXLMS, which converges slowly when experiencing large plant model errors. However, the properties of the algorithms control signal prove to be a little strange, especially when there is inaccuracy in the plant model. The leakage term that ensures that the algorithms command phase aligns with that of the disturbance signal causes the control effort to oscillate from maximum to minimum over time, as discussed in appendix A. This is not shown in any of the figures, as it occurs over several thousand iterations. However, even though the average control effort is still lower than the command-FXLMS, the magnitude-controlled FXLMS would still require loudspeakers of an equal power rating to the command-FXLMS to cope when the control effort is at a maximum.

SOSO-FXLMS

Although the SOSO-FXLMS can not be directly compared to the other algorithms in this report, as it cannot fulfil both phase and magnitude output criteria, it is useful to see how a sum of squared outputs approach compares to the best algorithms that use criteria $\{1\}$. The simulations run, using the SOSO-FXLMS, show that the algorithm allows both considerably faster convergence and a considerable reduction in control effort, due to the flexibility of the constraint put upon the control signals. It would seem from this initial investigation of the SOSO-FXLMS that, if a mean level across all microphones was acceptable within the car cabin rather than individual levels and phases at individual microphones, this method far out-performs all the other algorithms.

Conclusions

The aim of this report was to investigate the relative merits and failings of both new and existing sound profiling algorithms, and there performance criteria, for use in a vehicle sound profiling control unit. The algorithms were tested in terms of the amount of control effort used and their stability to plant model phase errors. These two properties are of key importance to the functioning of the control system. Obviously, the stability of the system dictates whether the algorithm can retain control or not. The effect of control effort is much more subtle and is

a measure of how hard the control system has to work, proportional to the power required to be delivered by the control loudspeakers. The amount of control effort can vary substantially and is dependent on the relative phases of the disturbance, control and target output signals. Thus, depending on whether or not the algorithm has enough information on these signals will affect the algorithms ability to minimise the amount of control effort used to achieve a given output. Whether an algorithm requires a large amount of control effort or not can effect the hardware that is required in the control system and it is thus advantageous to keep the control effort to a minimum. The algorithms were tested in simulations that model the extremes of sound profiling in a multichannel control system.

From the simulations conducted in this report it can be concluded that for fulfilling performance criteria {1}, i.e. where both the phase and the magnitude of the output signal are relevant, the APC-FXLMS seems the most appropriate algorithm to be used in the control unit. The algorithm, in it's current form, does not decrease the required control effort by much compared to the command-FXLMS, but with a degree of tweaking this control effort saving could be increased. More importantly, the only algorithm that provides control with less control effort is the PSC-FXLMS, which has been proven to not be stable enough at the high system gains required for the sound profiling in the C-MAX.

It has been shown that if the performance criteria is changed, such that only the sum of squared outputs is considered rather than individual signal properties, as in criteria {3}, great control effort savings using the SOSO-FXLMS algorithm can be achieved. The important question to ask, however, is whether or not an average sound field is acceptable within the car cabin. The answer to this question is subjective in nature and it is important that tests are carried out to discover the impact, if any, that an averaged sound field has on the sound quality within the car.

Appendix A

The Development of A Magnitude-Controlled Adaptive Algorithm

In this appendix two new algorithms based on the learning algorithm known as *Oja's Rule* [3], taken from principal component analysis, are proposed to be used to update the adaptive filter weights of a feedforward control system such that any predetermined value can be obtained under the performance criteria {2} and {3}, with the smallest amount of control effort possible. Both algorithms proposed converge on the target output in a different way to previous FXLMS-based algorithms, by having a positive convergence coefficient, which makes the algorithm numerically unstable, but by using other controlling terms the algorithms stability can be retained.

The "Oja's Rule" Algorithm (Oja's LMS)

This algorithm was originally designed to be used for the principal component analysis of neural networks, in which it was stated by Hebb [4] that *" when an axon of cell A is near enough to excite a cell B and repeatedly and persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased"*. This statement gave birth to the Hebbian learning algorithm, which describes how the output of an axon, $y(n)$, changes over time, n, with an input firing the axon, $x(n)$, by changing a weight vector, $\mathbf{w}$, which describes the transfer function between the input and output, such that $y(n) = \mathbf{w}^{\mathrm{T}}\mathbf{x}(n)$. The weight vector adaption is thus described by the equation

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \beta(n)[y(n)\mathbf{x}(n)] \tag{A.1}$$

where $\beta(n)$ is a step-size parameter. If left unchecked such a system is boundless and numerically unstable, which lead to the work of Oja [5], who proposed a normalised Hebbian rule, in which the weight vector is normalised at each iterative step, thus keeping the algorithm from going unstable. After normalisation and considering only the first order terms this changes

(A.1) to

$$\mathbf{w}(n+1) = \mathbf{w}(n)[1 - \beta(n)y^2(n)\mathbf{x}(n)] + \beta(n)[y(n)\mathbf{x}(n)] \tag{A.2}$$

It can be seen from (A.1), that the Hebbian rule bears a very close resemblance to the LMS algorithm, used in feedforward active control, except for the sign of the step-size parameter. It was further noticed that Oja's rule could in fact be used for the adaption of a weight vector to produce a desired output from a sinusoidal input signal and thus achieve active sound profiling. This, however, requires a small alteration to (A.2) as the output required is now equal to the difference between the output from the adaptive weights and an additional disturbance, $d(n)$, such that $y(n) = d(n) + \mathbf{w}^{\mathrm{T}}(n)\mathbf{x}(n)$. We now rewrite (A.2) as

$$\mathbf{w}(n+1) = \mathbf{w}(n)[1 - \alpha\rho u^{\mathrm{H}}(n)u(n)] + \alpha y(n)\mathbf{x}(n) \tag{A.3}$$

where the step-size, $\beta(n)$, has been replaced with the convergence coefficient α, to conform with active control notation. In addition the second term in the square brackets, that prevents the algorithm from going unstable and determines to which value the algorithm will finally converge upon, has been changed to now be dependent upon the control effort term $u^{\mathrm{H}}(n)u(n)$, where the output from the weights $u(n) = \mathbf{w}^{\mathrm{T}}(n)\mathbf{x}(n)$, and ρ is a parameter which determines the value of the final output $y(n)$. By controlling the value of ρ and the sign of α the algorithm can be made to adapt the weights such that the disturbance signal is either cancelled, attenuated or enhanced.

Determination of Control Effort Parameter ρ

The value of ρ is chosen such that $y(n)$ will converge to the same amplitude as a desired signal, known as the command signal, $c(n)$, such that criterion {2} is satisfied. Thus it was initially proposed that the control effort parameter were to be scheduled according to the equation

$$\rho = \rho' \frac{1}{||c(n)| - |y(n)||} \tag{A.4}$$

where ρ' is a constant that determines the rate at which ρ changes. In practice the magnitudes of $c(n)$ and $y(n)$ can be obtained from taking a running average of the two signals. As can be seen from (A.4), as $|y(n)|$ tends to $|c(n)|$ the value of ρ increases and thus slows down the adaption of the weight vector, hence the algorithm in theory converges towards the desired command amplitude, as in Fig. A.1. It is still important, however, to choose a suitable value for ρ'. For example if ρ' is too small when in enhancement mode (i.e. for positive α, negative α would be required for attenuation), as shown in Fig. A.2, the output exceeds the command amplitude. While, if ρ' is too large when in enhancement mode, as shown in Fig. A.3, the output never achieves the command amplitude.

In some cases, even when the value of ρ' seems to be chosen correctly, this can sometimes result in a sudden increase of the controlling term as $||c(n)| - |y(n)|| \to 0$, which results in a jump of the weight vector values, which causes the output $y(n)$ to bounce along the desired amplitude of $c(n)$, as shown in Fig. A.4. To fix this problem a small constant can be added to the denominator in (A.4), which stops ρ becoming to large. However, this then limits the accuracy to which the output can achieve the desired command amplitude, and in effect is the same as scheduling ρ to too large a value.

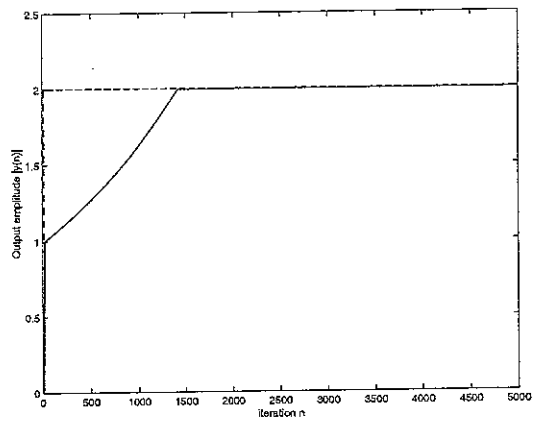


Figure A.1: Enhancement of output signal amplitude $|y(n)|$ to amplitude of $c(n)$ with $|d(n)| = 1$, $|c(n)| = 2$, $\alpha = 0.001$ and $\rho' = 0.001$.

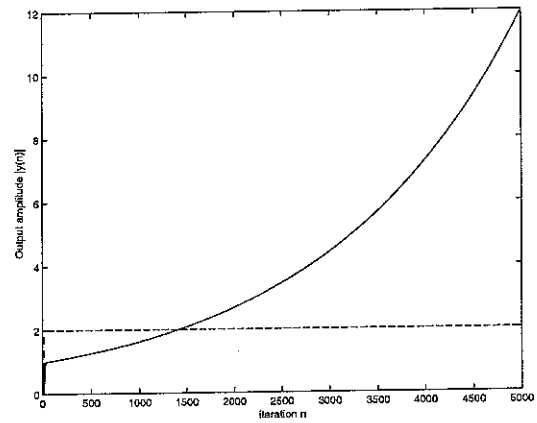


Figure A.2: Enhancement of output signal amplitude $|y(n)|$ to amplitude of $c(n)$ with $|d(n)| = 1$, $|c(n)| = 2$, $\alpha = 0.001$ and $\rho' = 0.0001$.

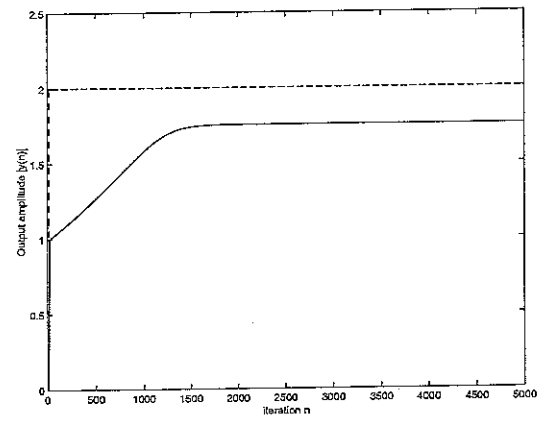


Figure A.3: Enhancement of output signal amplitude $|y(n)|$ to amplitude of $c(n)$ with $|d(n)| = 1$, $|c(n)| = 2$, $\alpha = 0.001$ and $\rho' = 1$.

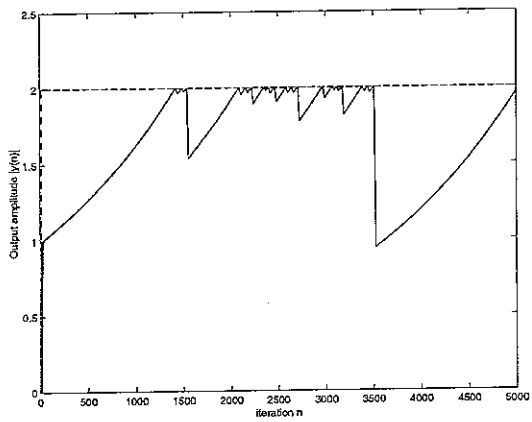


Figure A.4: Enhancement of output signal amplitude $|y(n)|$ to amplitude of $c(n)$ with $|d(n)| = 1$, $|c(n)| = 2$, $\alpha = 0.001$ and $\rho' = 0.005$.

Unfortunately, the choice of ρ' is not a simple task and depends on many factors. From (A.3) we can see that the adaption of the weights is not only dependent on the value of the convergence coefficient and the gradient of the cost function term, but also on ρ and $u(n)$. Unlike algorithms such as the command-FXLMS, ANE-LMS and PSC-FXLMS [1], Oja's algorithm does not use a pseudo error, which is the difference between the output and the desired signal, to produce the cost function surface. This has the benefit that it uses a minimum amount of control effort to produce the desired output, but also means that the gradient term has no direct correlation to the control effort. This means that if $|d(n)|$ is large, even if $|c(n)|$ is large, the gradient term will still be large, even though $|u(n)|$ is small, therefore a large value of ρ' is required to control the algorithm. Oppositely, if $|d(n)|$ is small and $|c(n)|$ large, then initially the gradient is small, and thus requires a small value of ρ' such that the algorithm does not stop adapting the weights to soon. It therefore seems that to select the correct value of ρ', knowledge of the disturbance signal amplitude is required. Furthermore, the use of large values of α, to increase convergence speed, further decreases the tolerance of an inaccurate scheduling of ρ'.

By running simulations over a range of values for all parameters, it can be concluded that the sensitivity of scheduling ρ to achieve the desired output amplitude has to be so precise that practical implementation of the algorithm is highly unlikely to work effectively.

The magnitude-controlled FXLMS

Continuing from the work presented on the use of *Oja's rule* [5], for the update of an adaptive filter for active sound profiling, this section shows the creation of a new FXLMS based algorithm, known as the *magnitude-controlled-FXLMS*, which also at times uses a positive convergence coefficient to achieve a desired output. The resulting algorithm is presented along with results of simulations showing the relative merits and failures of the algorithm. The use of the algorithm is shown as both a single channel and multichannel/multi-frequency active sound profiling controller, which is used exclusively as a frequency domain adaptive algorithm.

The Single Channel Frequency Domain magnitude-controlled FXLMS

The magnitude-controlled-FXLMS algorithm was developed to overcome the problems experienced by Oja's LMS algorithm. The first of these was to remove the need for an effort weighting parameter, who's value proved very difficult to determine to provide good convergence. Instead, it was decided that the convergence would be controlled by a varying step-size which would be dependent on the difference between the command magnitude level, $C = |c(n)|$, and the current output magnitude level, $|y(n)|$, complying with criterion {2}. This calculation of the output magnitude uses a moving time average, which requires a buffered output signal. The presence of this buffer removes any computational load advantage that a time-domain LMS approach may have over it's frequency domain counterpart, as updating the filter weights on a sample by sample basis would require a large degree of computational power. For this reason, it was chosen that the algorithm would be operated exclusively in the frequency-domain, which when controlling multiple tones will increase the computational efficiency. This requires the implementation of both an FFT and an IFFT, to transform the 'real-world' signals in and out of the frequency-domain. The FFT is calculated every N samples where N is the length of the FFT and is also the minimum number of samples required to construct one period of the lowest reference frequency. This means that the weights are only updated every N samples, but is an acceptable update rate to retain control. The resultant N point spectrum is then folded and normalised to give a useful $N/2$ point spectrum. The complex values are then taken from the spectral bins corresponding to the reference frequencies in question. For this reason, the wavelength of the reference frequencies chosen must be an integer multiple of the sampling frequency, f_s, ie $1/f_n$ must be an integer value, where f_n is the normalised frequency. Using these complex values, the magnitude of each output signal from each spectral bin can thus be determined using

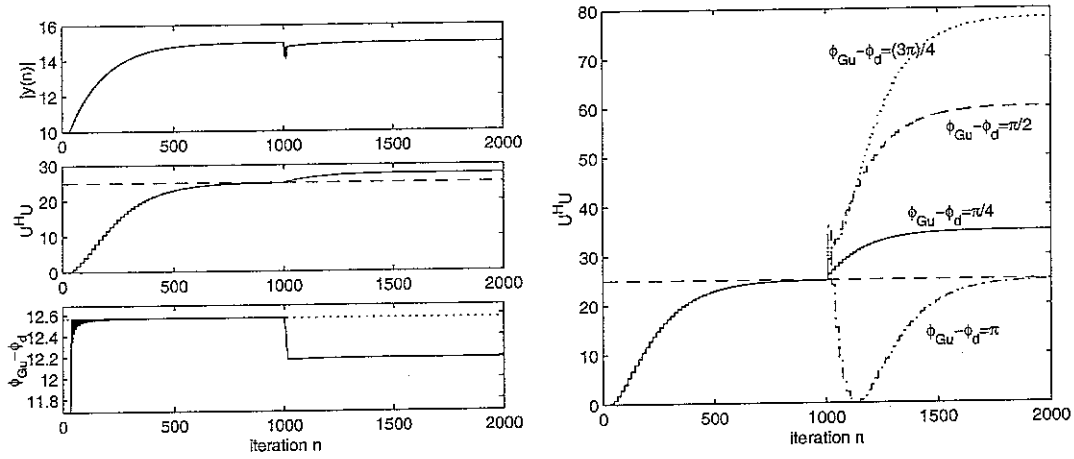
$$|y(n)| = \sqrt{Re\{y(n)\}^2 + Im\{y(n)\}^2} \tag{A.5}$$

In addition to this initial modification, the output magnitude is used to normalise the the output signal. This ensures that the step-size does not become to large at high output levels even though the required control signal level to achieve the desired output may be small, due to $C \sim |y(n)|$. Thus the update equation for the control signal for a single frequency is given by

$$u(n+1) = u(n) + \alpha \left(\frac{C - |y(n)|}{|y(n)|} \right) \hat{G}^* y(n) \tag{A.6}$$

where, assuming that the reference signal frequency has a magnitude of unity and has no phase relative to itself, $u(n+1)$ and $u(n)$ are the new and old complex control signal values, α is the convergence coefficient, $\hat{G}$ is the complex plant response at the reference frequency and $y(n)$ is the current complex output signal. In this section it will be assumed that all variables are complex, and are represented in the frequency-domain.

To reiterate, the aim of these algorithms is to produce the target output with the minimum amount of control effort possible. It was discovered, however, that if the algorithm has already converged upon an initial command level, i.e. $w(n) \neq 0$, and the disturbance signal phase is then changed, the minimum control effort is no longer retained, as shown in Fig. A.5(a). This effect occurs when the algorithm is in enhancement or attenuation mode, as both these modes have multiple solutions, to combinations of control signal phase and magnitude, to reach the desired output level. This increase in control effort occurs because the initial sum of the control



(a) Enhancement with Phaseshift=$\pi/8$

(b) Control Effort for varying phaseshift.

Figure A.5: Convergence of the magnitude-controlled-FXLMS in enhancement modes with an additional phase shift of $\pi/8$ in the disturbance signal at $n = 1000$. $|d(n)| = 10$, $C = 15$, $\alpha = 0.1$, $f_n = 1/16$, $N = 16$ and $\hat{G}(z) = G(z) = 1$. The dashed line shows the minimum control effort achievable.

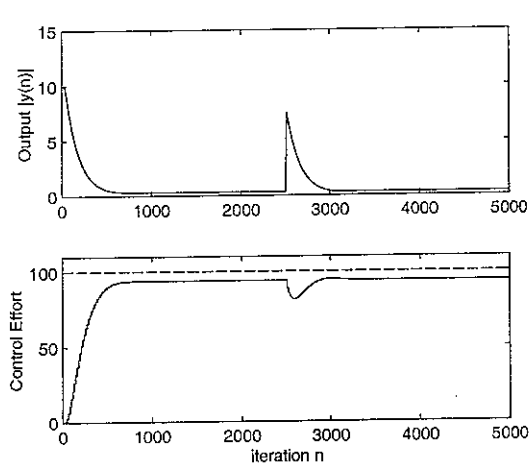
signal and disturbance signal, after the phaseshift, is no longer equal to the desired output level. This directly affects the gradient term of the update equation by increasing the value of the step-size. To correct for this phaseshift, only the phase of the control signal needs to be altered to regain the desired output level with the minimum control effort, yet the magnitude term also increases. Hence, by the time the phase has reached the value at which the desired output is achieved, the magnitude of the control signal is no longer at it's minimum value. Expectedly, the degree to which the control effort exceeds the minimum value is dependent on the size of the disturbance phaseshift. It can be seen from Fig. A.5(b) that this maximum control effort occurs when the phaseshift equal to $3/4\ \pi$.

It was realised that to fulfil the convergence criteria with a minimum of control effort, as with Oja's LMS, a leakage factor term had to be introduced into the algorithms update equation. Thus becoming

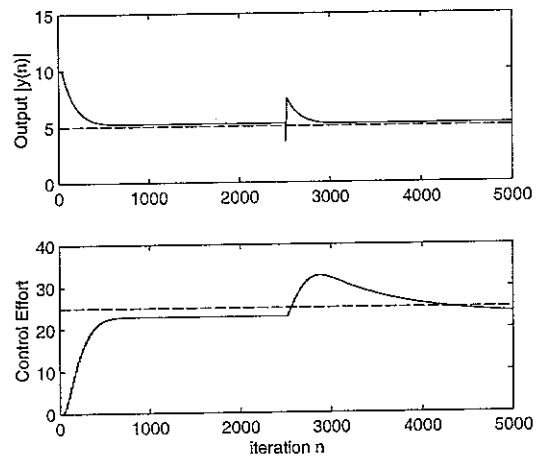
$$u(n+1) = (1-\rho)u(n) + \alpha\left(\frac{C-|y(n)|}{|y(n)|}\right)\hat{G}^* y(n) \tag{A.7}$$

where ρ is the leakage factor, which can either be a constant or a value dependent on another variable such as control effort. This leakage term subtracts from the value of the weight, thus allowing the gradient term to never reach zero and allowing the algorithm to cycle through all values of magnitude and phase to find those that correspond to the minimum control effort. In addition it was decided that the leakage term would be made to be dependent on the difference between the command level, C, and the output level, $|y(n)|$. This puts less constraint on the choice of the constant ρ, as the leakage term is now self-regulating and the output signal normalised, thus the update equation becomes

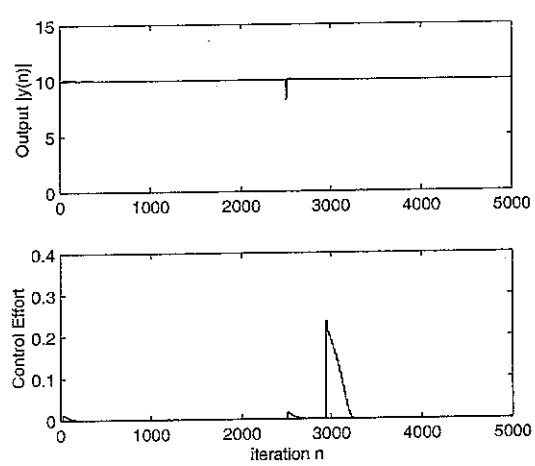
$$u(n+1) = \left(1 - \left|\frac{\rho}{C-|y(n)|}\right|\right)u(n) + \alpha\left(\frac{C-|y(n)|}{|y(n)|}\right)\hat{G}^* y(n) \tag{A.8}$$



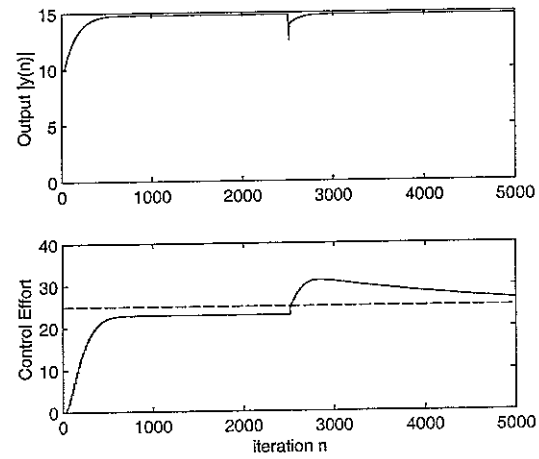
(a) Cancellation ($C = 0$)



(b) Attenuation ($C = 5$)



(c) Neutral mode ($C = 10$)



(d) Enhancement ($C = 15$)

Figure A.6: Convergence of the magnitude-controlled-FXLMS in all four sound profiling modes. $|d(n)| = 10$, $\alpha = 0.1$, $\rho = 0.001$, $f_n = 1/16$, $N = 16$ and $\hat{G}(z) = G(z)$. The phase of the disturbance signal was changed at $n = 2500$. The dashed line shows the minimum control effort achievable.

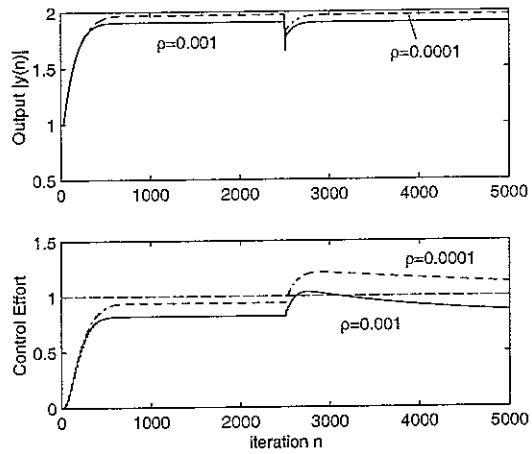


Figure A.7: Convergence of the magnitude-controlled-FXLMS in enhancement mode for different values of ρ. $|d(n)| = 1$, $C = 2$, $\alpha = 0.1$, $f_n = 1/16$, $N = 16$ and $\hat{G}(z) = G(z)$. The dashed line shows the minimum control effort achievable.

Fig. A.6 shows the convergence of the magnitude-controlled-FXLMS in all four active sound profiling modes. It can be seen from the plots that the algorithm possesses a dual convergence, once for the output and once for the control effort, i.e. the control signal will still continue to converge towards the minimum control effort without affecting the output, once the output has converged to the target level. By increasing the value of ρ the rate at which the algorithm converges on the minimum control effort is increased, while the accuracy to which the output can achieve to that of the target output is reduced, which can be seen from Fig. A.7 for $\rho = 0.001$ and 0.0001 in enhancement mode. In addition the rate at which the algorithm converges on the minimum control effort is dependent upon the degree of enhancement or attenuation, i.e. the greater the value of C/D the longer the algorithm takes to achieve minimum control effort, even though the desired output may have been achieved already.

The Effect of Plant Model Errors

The magnitude-controlled-FXLMS at an initial glance seems to be governed by the same parameters for stability as that of the standard FXLMS. This includes the condition that for stable convergence the plant model and physical plant must be within $\pm 90°$ phase of each other. As shown in Fig. A.6, once the algorithm is allowed to converge fully, the control effort required to achieve the desired output is at a minimum, provided the plant model is an accurate estimation to that of the physical plant, i.e. $\hat{G}(z) = G(z)$. As with all other FXLMS based algorithms, if $\hat{G}(z) \neq G(z)$ then the control signal will not be perfectly in or out of phase with the disturbance signal and the control effort required for convergence will increase above the minimum, as shown in Fig. A.8. However, errors in the plant model also cause another unwanted effect in the control effort. If the phase error and system gain, $C/|d(n)|$, are large enough, once the control effort has converged on the minimum value the filter weights continue to adapt. This results in the perpetual increase of the control signal phase, which causes the control to increase once more, see Fig. A.9. As long as the control signal is not adapting too quickly, this does not affect the output signal, which remains converged and stable. The rate

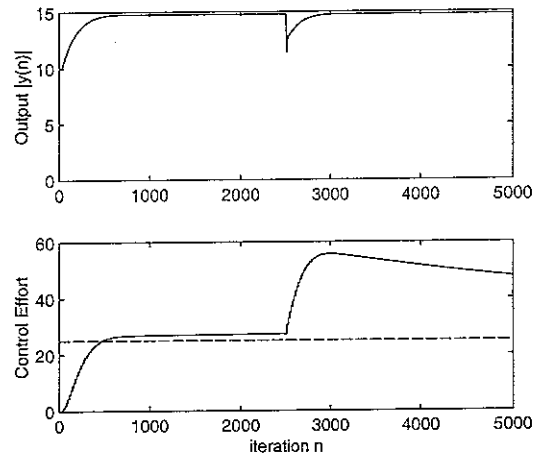


Figure A.8: Convergence of the magnitude-controlled-FXLMS in enhancement mode for an inaccurate plant model. $|d(n)| = 10$, $C = 15$, $\alpha = 0.1$, $\rho = 0.001$, $f_n = 1/16$, $N = 16$, $\hat{G}(z)/G(z) = z^{-1} \approx 22°$. This can be compared to the case where $\hat{G}(z) = G(z)$ in Fig. A.6(d). The dashed line shows the minimum control effort achievable.

at which the control effort oscillates is dependent on the value of ρ, the system gain and the degree of plant model phase error.

The Multichannel Channel Frequency Domain magnitude-controlled FXLMS

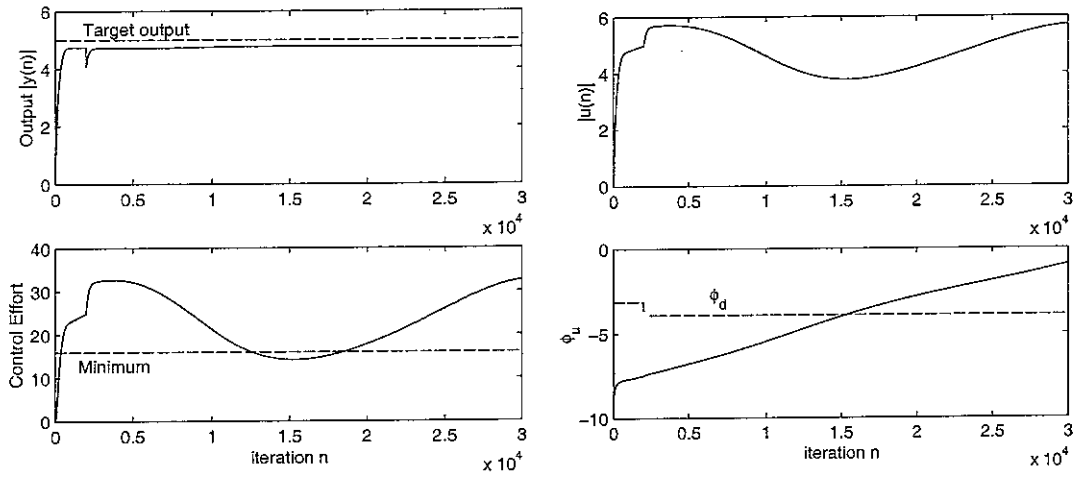
The multi-channel version of the magnitude-controlled-FXLMS is essentially the same as that of the single-channel version, however when changing the update equation from single- to multichannel, care needs to be taken when changing the scalar variables in (A.6) to vector and matrix variables. This applies specifically to the normalisation of the output vector, $\mathbf{y}(n)$, which must perform an element-wise division with the vector of output magnitudes. This can be represented instead in matrix form, by taking the inverse of a diagonal matrix who's non-zero elements contain the magnitudes of the output signals, such that

$$\mathbf{Y} = \begin{bmatrix} |y_1(n)| & 0 & \cdots & 0 \\ 0 & |y_2(n)| & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & |y_k(n)| \end{bmatrix} \tag{A.9}$$

where k is the number of sensors used in the system. The update equation is now written for a multichannel system as

$$\mathbf{u}(n+1) = \left(1 - \left|\rho[\mathbf{C} - \mathbf{Y}]^{-1}\right|\right)\mathbf{u}(n) + \alpha[\mathbf{C} - \mathbf{Y}]\mathbf{Y}^{-1}\hat{\mathbf{G}}^{\,\mathrm{H}}\mathbf{y}(n) \tag{A.10}$$

where $\mathbf{u}(n+1)$ and $\mathbf{u}(n)$ are the new and previous complex control signal vectors, $\mathbf{C}$ is a diagonal matrix of command level values of the same form as $\mathbf{Y}$ and $\hat{\mathbf{G}}$ is a matrix of complex plant model responses between sources and sensors, as shown in Fig. 2 for a 2 source/2 sensor system. For the multi-channel simulations shown in this report the physical plant was



(a) Output and control effort.

(b) Control signal magnitude and phase.

Figure A.9: Wandering effect of the magnitude-controlled-FXLMS in enhancement mode for $|d(n)| = 1$, $C = 5$, $f_n = 1/16$, $\alpha = 0.1$, $\rho = 0.001$ and $\hat{G}(z)/G(z) = z^{-2}$.

generated using the same format as on page 259 from Elliott's *Signal Processing for Active Control*, which is given by

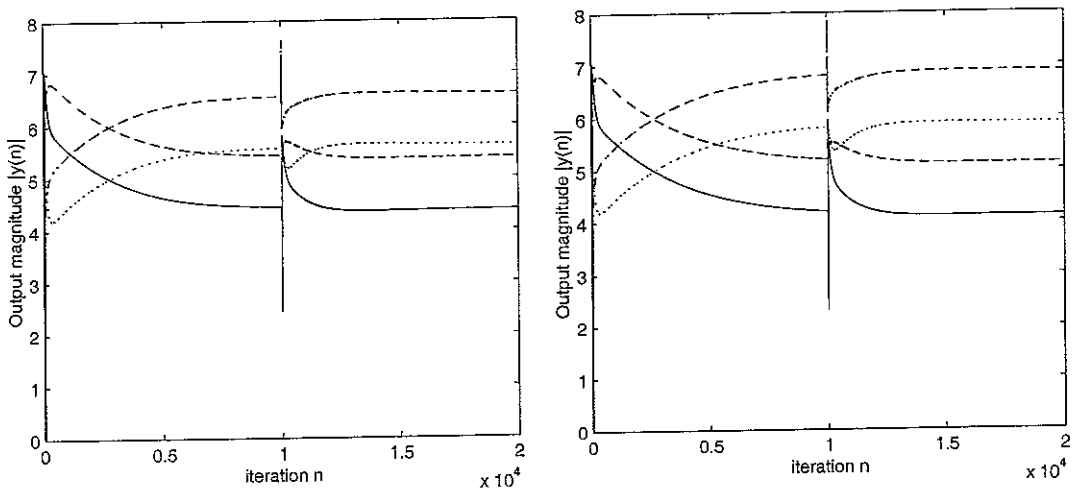
$$\mathbf{G}(j\omega) = \begin{bmatrix} \frac{A}{l_{11}} e^{-jkl_{11}} & \frac{A}{l_{12}} e^{-jkl_{12}} \\ \frac{A}{l_{21}} e^{-jkl_{21}} & \frac{A}{l_{22}} e^{-jkl_{22}} \end{bmatrix} \tag{A.11}$$

In this report A is set equal to 1. Although this frequency-domain approach to the plant does not give any transient response information to the plant, it is still a valid approximation of reproducing the physical plant.

Assuming the algorithm is stable and the system is well-defined, i.e. the number of sources is equal to the number of sensors, the magnitude-controlled-FXLMS has the ability to produce any output at any of the sensors and enables the soundfield to be tailored to a specific shape, thus satisfying criterion {2}, as shown for two different tones in Fig. A.10. It can be seen from the figure that the algorithm is perfectly capable of performing active sound profiling on such a 2-by-2 multichannel/multi-frequency system. In addition, the figure shows two plots for different values of ρ. It can be clearly seen from these two plots that, as discussed earlier, the greater the value of ρ the greater the inaccuracy of the convergence to the desired output.

Minimum Control Effort with No Phase Criteria

It can be shown that the minimum effort achievable by the magnitude-controlled FXLMS for a given target output to satisfy criterion {2}, both for single channel and multichannel systems, is equal to the minimum control effort by an algorithm that satisfies criterion {1}, but possesses the ability to schedule the phase of the command signal to that for the disturbance signal's,



(a) $\rho = 0.001$ (b) $\rho = 0.0001$

Figure A.10: Convergence of the multichannel/multi-frequency magnitude-controlled-FXLMS in enhancement mode for two different leakage factors of ρ. $\mathbf{d}_1 = [7\ 5]$, $\mathbf{d}_2 = [6\ 4]$, $\mathbf{c}_1 = [4\ 6]$, $\mathbf{c}_2 = [5\ 7]$, $f_1 = 1$ Hz, $f_2 = 2$ Hz, $f_s = 32$ Hz, $N = 32$ and $\hat{G}(z) = G(z)$, where $l_{11} = l_{22}$ and $l_{12} = l_{21}$ i.e. a symmetric plant. The vectors $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{c}_1$ and $\mathbf{c}_2$ contain the non-zero elements of the disturbance and command signal levels, respectively for each reference frequency.

as achievable by the PSC- and APC-FXLMS algorithms [1]. This can be shown analytically by considering a 2 microphone/2 loudspeaker multichannel single frequency system. Let the complex output signal equal

$$\mathbf{y}(n) = \mathbf{d}(n) + \mathbf{G}^{\mathrm{H}}\mathbf{u}(n) \tag{A.12}$$

which, when both the output and control effort have converged, is equal as a steady state solution to

$$\mathbf{c} = \mathbf{d} + \mathbf{G}^{\mathrm{H}}\mathbf{u}_{\mathrm{opt}} \tag{A.13}$$

If we only consider the solution to criterion $\{2\}$, then we are no longer concerned with the phase of the command signal, thus we only search for the solution to

$$|\mathbf{c}| = |\mathbf{d} + \mathbf{G}^{\mathrm{H}}\mathbf{u}_{\mathrm{opt}}| \tag{A.14}$$

If we now consider the case where we satisfy criterion $\{1\}$, where the phase of the command signal is now relevant, we can write (A.13) in terms of magnitude and phase, such that

$$\begin{bmatrix} C_1 e^{j\phi_{c1}} \\ C_2 e^{j\phi_{c2}} \end{bmatrix} = \begin{bmatrix} D_1 e^{j\phi_{d1}} \\ D_2 e^{j\phi_{d2}} \end{bmatrix} + \begin{bmatrix} G_{11}e^{-j\phi_{G11}} & G_{12}e^{-j\phi_{G12}} \\ G_{21}e^{-j\phi_{G21}} & G_{22}e^{-j\phi_{G22}} \end{bmatrix} \begin{bmatrix} U_1 e^{j\phi_{u1}} \\ U_2 e^{j\phi_{u2}} \end{bmatrix} \tag{A.15}$$

If the algorithm has the ability to schedule the phase of the command signal to that of the disturbance signal's then we can assume that $\phi_{c1} = \phi_{d1}$ and $\phi_{c2} = \phi_{d2}$, and similarly that

$$\angle[G_{11}U_1 e^{+j(\phi_{u1}-\phi_{G11})} + G_{12}U_2 e^{+j(\phi_{u2}-\phi_{G12})}] = \phi_{c1} \tag{A.16}$$

and

$$\angle[G_{21}U_1 e^{+j(\phi_{u2}-\phi_{G21})} + G_{22}U_2 e^{+j(\phi_{u2}-\phi_{G22})}] = \phi_{c2} \tag{A.17}$$

Both sides of (A.15) now have the same phase, which means that the magnitude of the sum of all terms in (A.15) is equal to the sum of all magnitudes of the terms in (A.15). Therefore, if all phase information is ignored from (A.15) then the magnitudes of $\mathbf{u}_{opt}$ for criteria {1} and {2} are equal, and thus the control efforts are the same. This proof shows that instead of using an algorithm that tries to predict the phase of the disturbance signal, like the PSC- and APC-FXLMS, an algorithm such as the magnitude-controlled FXLMS could be used instead. Which could eliminate the need for an internal model [1] and an FFT to conduct the algorithm in the frequency domain. Although, the magnitude-controlled FXLMS still requires a running time-average to calculate the magnitude of the output signal, which would probably show that the computational saving is so small to be of no consequence.

Conclusion

This report proposes two new algorithms for use in a feedforward active sound profiler. It was shown that from the two algorithms, the magnitude-controlled FXLMS could be suitable for both single- and multichannel control, while Oja's LMS required adjustment to such a fine degree as to be useless. In addition it was found that the magnitude-controlled FXLMS, designed to satisfy only magnitude criteria at the error microphones, uses the same steady-state control effort as a magnitude and phase criteria algorithm if the command signal phase is equal to that of the disturbance signal. It can be seen in this short report that the magnitude-controlled-FXLMS is an algorithm that is capable of performing multichannel active sound profiling and by far out performs the Oja-LMS algorithm, discussed in the first section of this report. However, it is unclear how the algorithm will perform in real-time experiments and compare to other active sound profiling algorithms. It is suspected that the success of the algorithm will be dependent on whether or not applications are tolerant to the algorithms output inaccuracy and or slow convergence to the minimum control effort, which may prove to be useless in the reduction of dependence on the power rating of the loudspeakers chosen for the control system.

Appendix B

The Optimised APC-FXLMS

It has been previously shown that although the APC-FXLMS has increased stability to plant model phase errors over the PSC-FXLMS, it does so at the expense of increased control effort. In fact in many cases the control effort advantage that the APC-FXLMS has over the command-FXLMS can be marginal. This raises questions of whether the additional computation that is required for the APC-FXLMS is worth it or not. It has been discovered, however, that the APC-FXLMS can be slightly altered to produce much improved control effort properties.

The coefficient '2', in (7), ensures that there is a smooth transition from enhancement to attenuation, where the APC-FXLMS acts like the PSC-FXLMS in attenuation mode. However, the APC-FXLMS can be altered such that the coefficient is changed to $1+\beta$, and thus (7) becomes

$$\phi_c = (1+\beta)\frac{\hat{D}}{C+\hat{D}}\phi_{\hat{d}} = \frac{1+\beta}{1+C/\hat{D}}\phi_{\hat{d}} \tag{B.1}$$

where β is the predicted enhancement gain, C/D, based on measured and target data. From (B.1), it can be seen that once the algorithm has converged and assuming that the plant model and estimated enhancement gain are accurate, the numerator and denominator of the fraction will be equal and thus the command and disturbance phases will be equal and the control effort will be a minimum. Interestingly, when the plant model is not accurate the algorithm has almost the same stability as when using the original phase scheduling in (7). This stability is retained because the value of $\hat{D}$ rises when the plant model is inaccurate and thus the scheduling of ϕ_c is reduced. As with the original algorithm, when $\phi_c \neq \phi_d$, the control effort increases above the minimum, which will increase as the actual enhancement gain moves towards neutral mode or to gains above that predicted.

There is, however, a hidden form of instability, which is dependent on the inaccuracy of the predicted enhancement. If the predicted gain enhancement is much larger than that of the actual enhancement gain then ϕ_c can become very large and, although it is not guaranteed, can send the algorithm into a limit cycle. Although this instability only arises when the enhancement gain estimate is extremely inaccurate. Albeit, if the actual enhancement gain is within close enough proximity of the predicted enhancement gain then the APC-FXLMS will

perform like the PSC-FXLMS but with far superior stability and achieve gains in excess of the PSC-FXLMS' stability.

Appendix C

Geometry of a Sound Profiling Cost Function Surface

The cost function surface is a geometrical representation of a desired cost function with respect to the values of the adaptive filter weight coefficients. In the case of single channel sound profiling problem only two filter weights are required, which can be represented by a single complex weight value. As only two variables are required to represent all the filter weights we can plot the cost function as a 3-dimensional image with the x- and y-axes representing the real and imaginary components of W, and the z-axis as the cost function. In a standard active noise control system that use the FXLMS algorithm to cancel the disturbance signal, ignoring the effect of the plant the cost function for a single channel, single frequency system is given by

$$J = e^*e = (d+u)^*(d+u) \tag{C.1}$$

where e is the error signal between the measured output and the desired output, d is the disturbance signal, u is the control signal and c is the command signal, where all variables are complex and steady state and $\{\ \}^*$ denotes the complex conjugate. The resulting cost function surface is hyper-parabolic and is shown in Fig. C.1. As can be seen from Fig. C.1, this surface only has a single point that represents the minimum value of the cost function. This means that the algorithm, which will only come to rest when the gradient of the surface is equal to zero, will always find filter weight values that correspond to the minimum value of the cost function, provided satble convergence.

In the case of a sound profiling system, as discussed at the beginning of the report, there are several output criteria that can be fulfilled by such a system. The choice of a specific output criteria can alter the position and shape of the cost function surface, which can in turn change the location of the minimum value of the cost function. In the case of criteria $\{1\}$, where both the magnitude and the phase of the output signal are defined, the shape of the cost function, although moved, remains hyper-parabolic, with a single minimum, the same as the active noise control cost function. However, if criteria $\{2\}$ is chosen, where only the magnitude of the output signal is defined, but with no concerns for the phase of the output, the shape of the cost function surface is no longer hyper-parabolic. For criteria $\{2\}$ the cost function for a

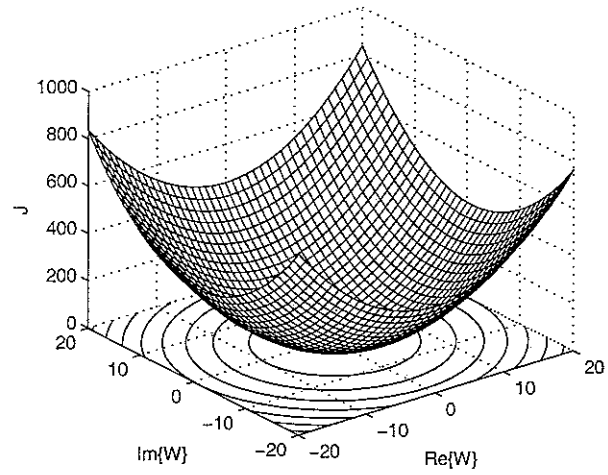


Figure C.1: 3-dimensional plot of the cost function surface for active noise cancellation, where $d = e^{(j\omega_r + \pi/2)}$.

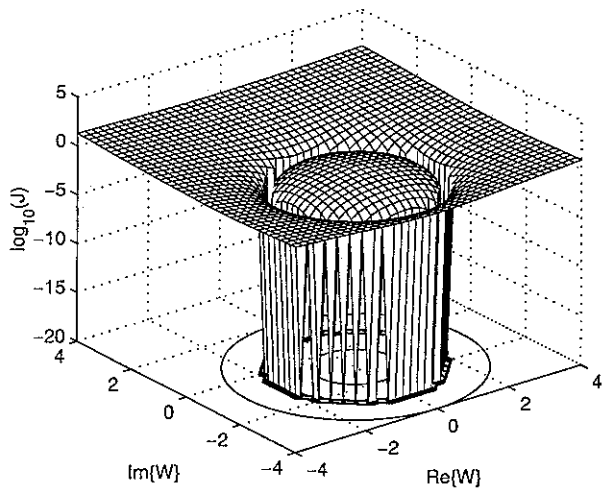


Figure C.2: 3-dimensional plot of the cost function surface for magnitude controlled active sound profiling, where $|c| = 2$ and $|d| = 1$.

single channel can be described as

$$J = e^2 = (|d + u| - |c|)^2 \tag{C.2}$$

where the complex conjugates are no longer necessary as all values are real. The cost function described by (C.2) is plotted in Fig. C.2. As can be seen from the figure, the cost function for criteria {2}, although shown on a logarithmic scale, is no longer parabolic with a single point minimum, but now has a minima which is ring shaped. All points on this ring are of equal value and can thus be the solution to criteria {2}. This geometry of the cost function, is what makes the idea of criteria {2} so appealing, as the ring minima has the advantage of being obtained via any number of combinations of phase and magnitude. This reduces the constraints on the algorithm both in terms of convergence time and stability. However, the shape of the cost function also brings about an inherent problem when trying to achieve the minimum control effort. As all points on the ring are of equal value, once a point on the ring

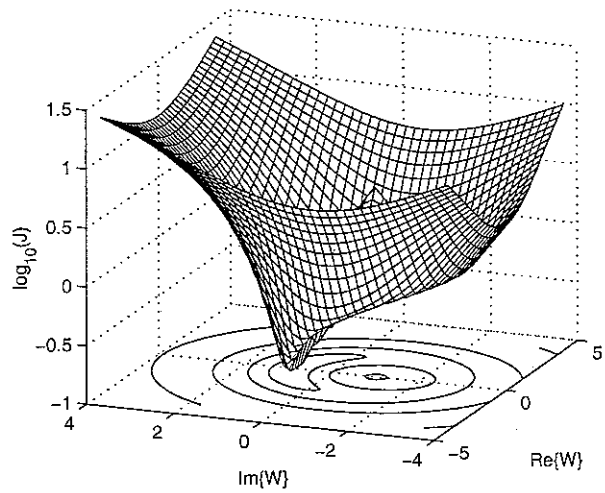


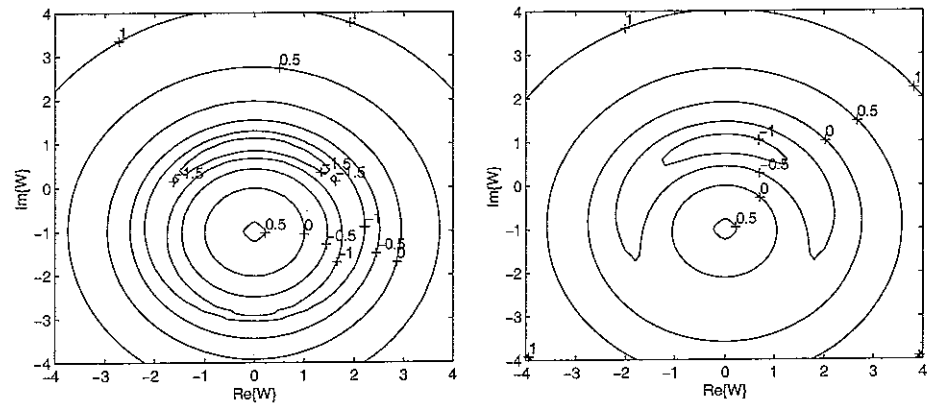
Figure C.3: 3-dimensional plot of the cost function surfacefor magnitude controlled active sound profiling with control effort, where $c = 2$, $d = 1$ and $\rho = 0.1$.

is reached a standard gradient descent algorithm will stop adapting, as all points on the ring have zero gradient. This means that although one solution may be achievable by using the minimum amount of control effort (defined by the point on the ring closest to the origin), unless the algorithm has already achieved this value, it will never be able to. The solution to this problem is to some how distort the cost function surface such that it only has a single minimum, which is located at the point of minimum control effort.

One way of changing the shape of the cost function is by introducing an additional term to the cost function in (C.2). If this term was proportional to the control effort then the minimum control effort and the minimum of the cost function would be co-located. An example of such a cost function can be derived by simply adding a term to (C.2), as given by

$$J = (|d + u| - |c|)^2 + \rho u^* u \tag{C.3}$$

where ρ is known as the control effort parameter, which defines the contribution of the control effort term u^*u to the cost function. If the new cost function is now plotted, as shown in Fig. C.3, it can be seen that there is only one point that defines the global minimum. Provided the algorithm is given enough time to converge upon the global minimum, using this cost function both the target output and the minimum control effort will be achieved. Unfortunately, this is not entirely true. By distorting the shape of the cost function surface, by the addition of the control effort term, the value of the filter weights at the minimum are also distorted. This means that when the algorithm reaches the global minimum of the cost function surface the value of the weights no longer exactly produce the target output. The contribution of the control effort term in (C.3) depends on the distance from the origin in the weight plane and thus, unless the amount of control effort required to achieve control is equal to zero, which would apply no control, the cost function minimum will never exactly result in the production of the target output. However, the difference between the optimum filter weights, those that produce the target output, and those corresponding to the minimum of the cost function can be minimised. This is done by altering the value of the control effort parameter, ρ. By making the value of ρ small the contribution of the control effort term to



(a) $\rho = 0.01$ (b) $\rho = 0.05$

Figure C.4: Contour plots of the cost function surface. The two figures show the difference in surfaces for different values of control effort parameter ρ.

the cost function is decreased and thus the distortion and the error from the target output minimised. However, by decreasing the contribution of the control effort term, the gradient of the cost function surface towards the global minimum is reduced, as shown in Fig.C.4. It is clear to see from the figure that the plot of $\rho = 0.01$, Fig.C.4(a), has contours that are almost all circular, implying a very small amount of tilt to the cost function surface. In Fig. C.4(b), however, there is an increase in banana shaped contours that implies a greater tilt to the cost function surface, as in Fig. C.3. As discussed in the main text, with regards to the leakage term in the magnitude controlled FXLMS algorithm, this slows the convergence of the algorithm to the global minimum. The algorithm may have converged to the ring of equal output magnitude, but if the gradient across the ring is small the algorithm will descend slowly to the global minimum. This poses the question "what is more important speed or accuracy?" The answer is probably a balance between the two, but this will depend on the application and environment of the control system.

Appendix D

MATLAB Simulation Programs

The following section gives a brief description of the MATLAB programs used to generate the results and figures contained in this report.

allFDSCSF.m

This program simulates the active sound profiling of a single-channel single-frequency system. Some of the inputs include: disturbance signal frequency and amplitude, command signal amplitude, plant phase response, plant model phase response, selection of algorithms to use, convergence coefficient value and simulation run length. The program can output various plots including time series of the output signal magnitude and control effort.

all2by2SCFD.m

This program is the multichannel equivalent of allFDSCSF.m. It simulates a multichannel, single-frequency for a 2 sensor/2 source system. In addition to the allFDSCSF.m the locations of the sensors and sources can be input as 2-D coordinates, which then generates the corresponding plant response matrix between sources and sensors for an anechoic environment. The program outputs plots of both combined and individual output magnitudes and control effort.

all2by2MFFD.m

This program is the same as all2by2SCFD.m, except it simulates the control of two frequencies instead of one.

ASPcostfunction.m

This program plots the 3-D cost function surface of a complex filter weight for active sound profiling. Inputs include: disturbance magnitude, command magnitude, choice of performance criteria and value of control effort parameter. The program outputs a series of figures that show the cost function surface in a manner different ways including 3-D mesh plots and 2-D contour plots.

ojaLMS.m

This program simulates the active sound profiling of a single-channel single-frequency disturbance signal using the Oja's LMS algorithm, as described in appendix A. Inputs include: disturbance signal magnitude, command signal magnitude, reference frequency, sampling frequency, convergence coefficient value and control effort parameter.

References

[1] L. E. Rees and S. J. Elliott, "Adaptive algorithms for active sound-profiling." Accepted and to be published in *IEEE Transactions on Speech and Audio Processing*, March 2006, 2004.

[2] S. M. Kuo and M. J. Ji, "Development and analysis of an adaptive noise equaliser," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 217–222, May 1995.

[3] K. I. Diamantaras and S. Y. Kung, *Principal Component Neural Networks*. John Wiley & sons, 1996.

[4] D. O. Hebb, *The Organisation of Behaviour*. John Wiley & sons, 1949.

[5] E. Oja, "A simplified neuron model as a principal component analyser," *Journal of Mathematical Biology*, vol. 15, pp. 267–273, 1982.