TÉCNICO LISBOA

Active Noise Control Headphones

André Filipe Barroso Cerqueira

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor(s): Prof. Paulo Alexandre Crisóstomo Lopes

Examination Committee

Chairperson: Prof. Nuno Cavaco Gomes Horta
Supervisor: Prof. Paulo Alexandre Crisóstomo Lopes
Member of the Committee: Prof. João Pedro Castilho Pereira Santos Gomes

November 2016

To my family

Acknowledgments

First and foremost, I would like to thank my supervisor, Professor Paulo Lopes, for giving me the opportunity to work in a project so in tune with my interests, on this last step of my academic path. His guidance, technical support and patience throughout my dissertation were fundamental for its completion.

I would also like to thank INESC-ID for the financial support and for the use of their facilities. A big thank you for the lab mates who shared their experience and taught me a great deal.

I would like to thank my friends and girlfriend, Juliana, for all their help, love and care.

And to finish, I owe the biggest debt of gratitude to my parents, for supporting me all the time throughout the degree and encouraging me to complete the Master degree and the thesis.

Resumo

Este documento relata o projeto de um protótipo de auscultadores com controlo activo de ruído (CAR) adaptativo.

CAR pode ser usado com barreiras acústicas para atenuar ruídos de baixa frequência, complementado o fraco desempenho que estas têm ao cancelá-los. Controlo activo adaptativo tem a vantagem de conseguir eliminar sinais periódicos numa maior banda de frequências que as soluções comerciais baseadas em controlo activo fixo.

Durante o projecto, um par de auscultadores *ATH-ANC1 QuietPoint*^{TM} tiveram o seu controlador alterado para um controlador baseado numa uma placa de desenvolvimento C6713DSK que contém um TMS320C6713, um processador digital de sinal DSP de vírgula flutuante fabricado pela Texas Instruments (TI).

Vários algoritmos de feedback ou feedforward com modelação de caminho secundário offline ou online foram implementados. O caminho secundário do sistema foi caracterizado, o erro de fase da estimativa do caminho secundário foi avaliado e foram estudadas soluções ao problema. A relação entre o desempenho e parâmetros do sistema foram estudados experimentalmente de forma a optimizar o sistema.

Finalmente o desempenho no cancelamento de vários tipos de ruídos do sistema desenvolvido foi comparado ao de outros auscultadores ANC relatados na literatura. Ruídos tonais foram completamente cancelados, restando apenas o ruído ambiente. Ruído de banda estreita teve atenuação total de 7.1 dB, com componentes harmónicas canceladas em 23.41 dB, contudo ruído de banda larga não foi cancelado. Os resultados experimentais são aceitáveis uma vez que se comparam ou superam o desempenho dos sistemas comerciais relatados, contudo são inferiores aos resultados recentes de outros investigadores. O sistema apresenta problemas de estabilidade, que podem dificultar uma aplicação comercial, pelo que merecem ser melhor estudados.

Palavras-chave: Controlo de Ruído Activo, TMS320C6713, Modelação On-line, Modelação Off-line, Auscultadores Feedforward, Auscultadores Feedback, MMFxLMS, FxLMS, BMFxLMS

Abstract

This document reports the project of an adaptive active noise control (ANC) headset prototype.

ANC can be used to attenuate low frequency acoustical noises alongside passive barriers, to complement their poor performance when cancelling those noises. Adaptive active control is capable of cancelling periodic noise in a larger band of frequencies than commercial noise cancelling solutions, based on fixed active control.

During the project, a pair of *ATH-ANC1 QuietPoint*^{TM} headphones had their controller changed to a controller based on the C6713DSP starter kit (DSK) development board, housing a Texas Instruments (TI) TMS320C6713 floating point digital signal processor (DSP).

Various feedback or feedforward ANC algorithms with offline or online secondary path modelling were implemented. The system's secondary path was characterized and the phase error of secondary path estimate was assessed and solutions to correct it were studied. The relation between performance and system parameters was studied experimentally in order to optimize the system.

Finally the performance of the cancellation of various types of noises of the developed system was compared to the performance of other ANC headsets reported in the literature reviewed. Tonal noises were completely cancelled leaving only the background noise. Narrowband noise had a total attenuation of 7.1 dB, with harmonic components cancelled by 23.41 dB, however broadband noise cancellation was not achieved. The experimental results are acceptable since it compares, and sometimes it exceeds, the performance of the reported commercial systems, however if obtained inferior attenuation compared to recent results from other researchers. The system presents stability issues, which can difficult a commercial application, thus they should be better studied.

Keywords: Active noise control, TMS320C6713, On-line modelling, Off-line modelling, Feedforward Headset, Feedback Headset, MMFxLMS, FxLMS, BMFxLMS

Contents

List of Tables

List of Figures

Acronyms

ADC Analog-to-digital converter

ANC Active noise control

BMFxLMS Band limited MFxLMS

CCS Code Composer Studio

CODEC Coder-decoder

DAC Digital-to-analog converter

DSK DSP starter kit

DSP Digital signal processor

FIR Finite impulse response

FxLMS Filtered-x LMS

IIR Infinite impulse response

JFET Junction field effect transistor

LMS Least mean squares

McBSP Multi-channel buffered serial port

MFxLMS Modified FxLMS

MMFxLMS Mirrored MFxLMS

MSE Mean Square Error

NLMS Normalized least mean squares

OMA Overall modelling algorithm

OPAMP Operational amplifier

PCB Printed circuit board

SMD Surface-mounted device

THD Total harmonic distortion

List of Symbols

$*$	Convolution operator
$\hat{\mathbf{s}}(n)$	Coefficient vector of $\hat{S}(z)$
$\hat{d}(n)$	Primary noise Estimate
$\hat{S}(z)$	Estimate of secondary-path transfer function
$\hat{S}_*(z)$	Optimum Estimate of $\hat{S}(z)$
$\hat{y}'(n)$	Anti-noise estimate
λ	Leakage factor
μ	Step size
$\partial/\partial\mathbf{w}(n)$	Gradient of the mean square error surface
p	Crosscorrelation vector of the reference signal and the desired response
R	Autocorrelation matrix of the input signal
$\mathbf{w}(n)$	Tap-weight vector Coefficient vector of $W(z)$
$\mathbf{w}_*$	Optimum weight vector
x'(n)	Filtered-x vector
$\mathbf{x}(n)$	Tap-input vector Signal vector
$A(z)$	Feedback-path transfer function
$A_{V_{\text{pre}}}$	Error microphone pre-amplifier gain
$A_{V_{\text{ref}}}$	Reference microphone pre-amplifier gain
$C(n)$	Mean square error cost function
$d(n)$	Desired response Primary noise
$e(n)$	Error signal
$E[.]$	Expectation value operator

$e_m(n)$ Modified Error Signal

$F(z)$ Band-limiting filter transfer function

F_s Sampling frequency

Gd Digital gain of the training signal.

Gv Power amplifier gain

L Filter length / Number of taps

n time instance

$P(z)$ Primary-path transfer function

P_x Power of $x(n)$

$S(z)$ Secondary-path transfer function

t_p Processing time of noise cancelling algorithm.

T_s Sampling period

$W(z)$ Control filter transfer function

$w_l(n)$ Adaptive filter coefficient

$x'(n)$ Filtered reference signal in FxLMS

$x(n)$ Reference input signal Reference noise

$y'(n)$ Anti-noise

$y(n)$ Filter output Control Signal

Chapter 1

Introduction

1.1 ANC Overview

Acoustical noise may be viewed or defined as an unwanted or disturbing sound perceived by the human auditory system. Its rising levels are a widespread problem in society since in industrial, office, transportation and home environments there are motors, transformers, compressors and fans adding to the ambient noise [1]. This contributes to the degradation of our quality of living because prolonged exposure to excessive levels of acoustic noise can cause permanent hearing loss, safety problems, lower worker productivity and physical and psychological health issues [2].

Two types of noise may be considered according to their frequency content and time variance. Broadband noise, also called wideband or turbulent noise, is caused by turbulence. Since it is totally random it distributes its energy more or less evenly across the frequency band. Examples for this type of signal are the low frequency sounds of jet planes, impulse noise of an explosion and white noise. This kind of non-predictable signal is more difficult to control because it is not possible to cancel it in the absence of a coherent reference signal, measured in advance [3].

Narrowband noise is a periodic signal or a finite sum of periodic signals with its sound energy concentrated at specific frequencies. Narrowband noise is predictable and time invariant since it is caused by rotating machinery. Therefore, it can be effectively controlled with techniques which do not rely on causality (having prior knowledge of the noise signal) [4]. Some examples for its sources are internal combustion engines, compressors, vacuum pumps, turbines.

The traditional approach to acoustic noise control focus on preventing sound waves from reaching the eardrum using passive techniques [5]. This is achieved with enclosures, barriers, and silencers which absorb, reflect and disperse the sound waves reducing the energy of the disturbance before reaching the ears [6]. They have high attenuation over a broad frequency range but they are not effective in reducing low frequency noise. This happens because barriers need to have dimensions in the order of the noise wavelength which, for low frequency sounds, would imply a several meter barrier [7], an impractical and costly solution.

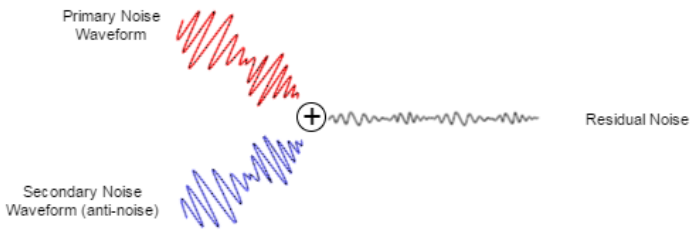


Figure 1.1: Active noise cancellation concept.

In an effort to improve noise reduction, active noise control (ANC)[1] has been proven to be an effective solution [3, 8]. ANC takes advantage of the destructive interference of waves [9], illustrated in Figure 1.1. It is explained with the principle of superposition of waves where, for a pair of superimposed waves, the resulting waveform depends on their frequency content, amplitude and relative phase. If the original wave and its inverse encounter at a junction at the same time, total cancellation occur.

In an ANC system the destructive interference is accomplished by generating a secondary noise field (anti-noise), and acoustically combining it with an unwanted noise field (primary noise) in an environment. Sensors are used to measure the primary noise and the residual signal, resulting from the interference, in order to obtain information about the system state. The signals are fed a control system that processes them and produces signals that drive loudspeakers (secondary sources), generating the anti-noise [1]. These fundamental elements are depicted in Figure 1.2.

The basic idea of active noise control was initiated in the 1930s by Paul Lueg [10], and more development was done in the 1950s by Harry Olsen and Everet May [11]. However, only with the development of digital signal processors (DSP), and the advancements of low-cost electronics and in control theory, in the late 1970s and 1980s, the research in active noise control experienced considerable progress and commercial interest. During the 1990s the first commercial products were marketed, even though with some resistance. After the turn of the century, several types of ANC systems became successful with the most popular being active headsets and earmuffs. At the end of 2009, active noise control and active vibration control systems, to reduce interior tonal noise in propeller driven aircraft, were successfully installed in operating commercial aircraft [12].

Air-acoustic ANC is most useful for low-frequency sounds where passive techniques are less effective, generally below about 500 Hz [9]. Sound fields at higher frequencies have complex sound fields increasing computational complexity for active methods. Since passive techniques accomplish good results at medium and high frequencies, both techniques should be used together to achieve better results in the whole audio band [13]. Thus, an attractive means to achieve large amounts of noise reduction in a small package with low weight and cost are active headsets.

There are two types of controller for an ANC system: fixed or adaptive. A fixed analogue controller falls under the field of classical control theory. They are realized by first modelling the environment where cancellation will occur, and then designing the optimal amplifier (electronic filter) for the desired

[1] also known as active noise cancellation and active noise reduction (ANR)

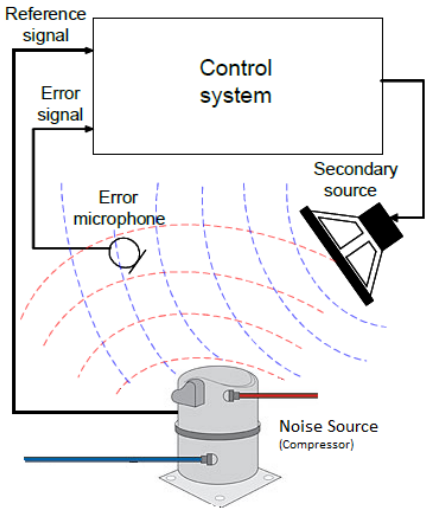


Figure 1.2: Basic blocks of a feedforward ANC system.

response. One of the ways to estimate the model is with an open loop measurement between the loudspeakers and sensors thus obtaining the frequency and phase responses of the model, i.e. the transfer function [14].

However, a fixed controller has several drawbacks. The major ones are being only applicable to linear time-invariant systems and hard to design for multiple input, multiple output systems [12]. Also, for most ANC applications, noise sources have time varying acoustic and environment characteristics, and the frequency content, amplitude, phase, and sound velocity of the undesired noise are non-stationary [1]. Further, obtaining an accurate model of an acoustic system and implementing a robust and stable analogue controller is an expensive and time consuming process. It requires high-quality microphones and speakers for the measurements and multiple iterations of filter design, testing and tweaking of parameters until a final product is achieved [15, 14].

For these reasons, the major application for non-adaptive control is ANC headsets, since they formed two separated single channel systems and where all the aforementioned properties are controlled through proper design. Fixed analogue controllers may have some advantages for this application such as stability, low system delay, low power consumption, small hardware size, simple control algorithm and good attenuation of broadband and impulse noises. However, headsets using this controller have cancellation over limited bandwidth for narrowband signals [3].

So, in order to track and respond to the aforementioned non-stationary characteristics in real time, the ANC system controller has to be adaptive. This is achieved using a class of adaptive systems from the field of signal processing, the adaptive filters. They are digital filters whose coefficients are adjusted to minimize an error signal (the desired signal minus the filtered signal) [4]. In contrast to the fixed controller, an estimate of the acoustical model, used by the controller to generate the anti-noise, is iteratively returned by the adaptive algorithm. The residual signal is sensed and input to the adaptive algorithm to improve the model. The error signal acts as a measurement of the deviation from the optimum solution.

Adaptive filters are implemented in digital signal processors which sample signals from electro-

acoustic transducers and process them in real time. They can be realized as (transversal) finite impulse response (FIR), (recursive) infinite impulse response (IIR), lattice, transform-domain filters and non-linear filters. The most common form of adaptive filter is the transversal filter using the least-mean-square (LMS) algorithm [1].

During the 1980s, the development of ANC received a boost in popularity due to the first theoretical adaptive cancellation systems being developed [16, 17] and because of the development of cheap and powerful DSPs. These DSPs were designed for very-high-speed numeric processing, which enabled the implementation of the powerful adaptive algorithms [4].

It has been demonstrated that the self-optimizing and tracking capabilities give the adaptive controller improved performance over fixed controllers, with narrow-band noise. Specifically, the adaptive controller can achieve 30dB or more[18] and a fixed controller a maximum of 30dB, depending on the headset design it can be less [19]. Another advantage is, as stated, the larger bandwidth of adaptive controllers over fixed controllers.

The ideal result of ANC is to achieve no noise, however real adaptive ANC systems leave a residual signal. To reduce this signal greatly the system must have a highly precise control, temporal stability and reliability [3]. This means that noise cancellation performance will depend on the system's ability to accurately estimate the primary noise characteristics at the junction point and synthesize its anti-noise with the correct amplitude and phase, every time requested, no matter the ambient conditions. So, the improvement of ANC lies on the development of improved adaptive signal processing algorithms, transducers, and DSP hardware. More sophisticated algorithms allow faster convergence, greater noise attenuation and better robustness to interference. The development of improved DSP hardware allow these algorithms to be implemented, improving system performance [1] and reducing size. However, there is a limit on the improvement of noise cancellation relying on hardware alone, so new techniques should be also researched.

Active noise control technology can be used in a wide variety of locations. These include car or airplane headrests, automobile exhaust mufflers and refrigerator fans. Headsets employing ANC would be particularly useful for airport ramp workers, ambulance drivers, and many others who operate on or near heavy machinery.

Headsets are one of the most convenient structures to apply ANC because these systems have a tendency to create small 'zones of cancellation' near the error sensor, which work best for small acoustic spaces, i.e. the ear canal. These and "silent zone" systems, employing multiple sensors and actuators (global cancellation), are the applications with more interest. An innovative product would be a hybrid active headsets which employed fixed controllers and adaptive controllers. The adaptive controller would reduce periodic signals while the fixed would control broadband signals. This is because the adaptive system is digital and the delay of the A/D and D/A converters are usually high, therefore analogue systems do a better job at cancelling broadband signals [20, 21, 22].

The contribution of this work is to further research adaptive controllers, by implementing, testing and comparing adaptive algorithms in a headset system. Also the impact of some implementation decisions on the system performance will be analysed.

1.2 Objectives

The goal of this work is the realization of a prototype for an ANC headset by stripping the commercial ANC headset *ATH-ANC1 QuietPoint*® of its control board and use the TMS320C6713 DSP starter kit (DSK) to develop an adaptive controller for the headset. It will act as a platform to experiment with different arrangements of ANC headset systems.

Robust and stable cancellation of broadband and narrowband noises will be aimed. To achieve that, different configurations of the system and algorithms will be tested and their performance assessed. Some of these algorithms come from research made at INESC-ID [23, 24], so it is expected to further that study with real time data of their operation. They will be compared with the more common filtered-x least mean squares (FxLMS) algorithm.

During implementation, a signal conditioning circuit for the sensors and an audio amplification circuit for the actuators will be developed. Also, signal processing functions and optimization algorithms will be written for the control system. During these steps, it is expected that these tasks give a better grasp on the design of analogue amplifiers, on the steps involved for the development of an electronic circuit prototype and on embedded systems' software design.

1.3 Thesis Outline

This document will provide an overview to ANC technology and its portable applications as it will offer a description of how the prototype was designed, built and tested.

Chapter 2 will present a summary of the concepts necessary to comprehend the development of the system arrangements. It starts with the presentation of fundamental concepts and terminology for ANC systems. Then their basic control structure, the adaptive filter, and a class of algorithms used in its implementation, the least mean squares (LMS), are introduced. Some important considerations to have with ANC systems and how they led to an important family of ANC algorithms, the filtered-x LMS(FxLMS), are talked about.

Chapter 3 will have a small review of the state-of-the-art on the ANC field, finishing with a few examples of portable applications recently researched.

After that chapter 4 will describe the system's hardware and software, justifying all design decisions made. It begins with a description of the system control board, the C6713 DSPstarter kit (DSK) and its daughtercard DSK_AUDIO4, is described in order to familiarize the reader with their overall capabilities. Then a thorough description of the entire analogue circuit, namely the microphones' biasing and pre-amplifiers, and speakers' drivers, is given from a circuit analysis point of view. Lastly the printed circuit board (PCB) design techniques are summarized. It also covers the firmware written for the DSP, with all the digital signal processing and noise cancelling algorithms explained and their C-language programs detailed.

Next the results from the system will be showed in chapter 5. First the secondary path of the system is characterized. Then a study of performance versus the system's parameters for each algorithm is

made. Lastly the various noise cancelling algorithms, with the best set of parameters, will have their performance evaluated and compared with literature results.

Lastly, this document will be concluded by presenting further options to improve this system and also offer an overview of what was globally accomplished with this system, facing the objectives that were set in the beginning.

Chapter 2

Background

This chapter does a small introduction to all of the important theoretical knowledge necessary to understand the project implementation and the results discussion. It is expected for the reader to be familiar with signals and systems, signal processing, filter theory and optimization.

2.1 Types of ANC systems

In this section concepts important for the physical description of an ANC systems are introduced. First the single-channel or multiple-channel systems classification is presented. It depends on the geometry of the sound field to control, which in turn depends on the acoustic space where the cancellation will occur.

Then feedforward and feedback system, the two configurations which ANC systems may be realized, are introduced. The systems are classified as such depending on whether or not the noise controller benefits of a separate reference input signal, acquired by a reference sensor, in an attempt to cancel the noise. It should be noted that the combination of both, hybrid systems, also exist.

2.1.1 Single-channel and Multiple-channel systems

One of the simplest applications of active noise control is associated with the attenuation of plane waves propagating in long narrow ducts. The simplicity of this application is a result of the sound field being restricted from spreading out in three dimensions by the walls of the duct and only propagating in one direction along the duct [12]. Therefore the problem is one-dimensional and a single-channel systems is used to cancel a sound field with those properties. The system has one cancelling source, one sensor to pick residual noise and one reference sensor (in feedforward structures) to pick a signal as noise reference. Also, these systems work best when used in small acoustic spaces since cancellation can only be achieved very close to the error microphone (about a volume of 1/10 of the wavelength) [12].

When a sound propagates in large ducts or enclosures, such as passenger compartments of vehicles, they display complex modal behaviour [3]. This means each of the noise harmonics will excite many room modes, complicating the geometry of the sound field, making the cancellation in such vol-

umes a multidimensional problem. To control these sound fields a multiple-channel system is used. It is composed of an array of sensors and actuators carefully placed in the enclosure. Before its implementation a careful investigation of the acoustic characteristics of the enclosure and sound field is required. Then, during design, the number of reference sensors, error sensors and actuators, and their properties as well as their position and size have to be considered. The system will also require the use of a multiple-input multiple-output adaptive algorithm.

2.1.2 Feedforward systems

A single-channel feedforward system, illustrated in Figure 2.1, relies on the availability of a reference signal, which is a measure of the incoming noise, to control the noise propagating in the duct. It consists of a reference sensor, a controller, generally digital, a loudspeaker and an error microphone.

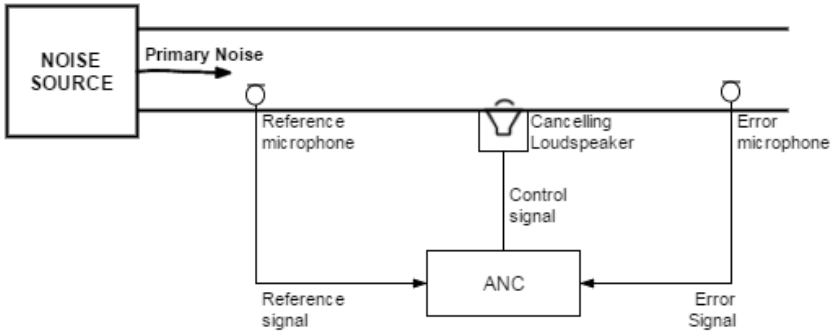


Figure 2.1: Single-channel feedforward ANC system in a duct.

On a feedforward system an undesired noise (primary noise), generated by a noise source, is picked up by a reference sensor as it travels down a duct. Then the controller modifies the reference signal and outputs a control signal. This signal is applied to a loudspeaker (secondary source), which creates an opposing pressure wave, 180° out of phase with the primary noise, when the noise wave reaches the loudspeaker. An error microphone monitors the residual acoustic pressure, which is used to adjust the controller for optimal results. The basic principle of the feedforward configuration is that the propagation time delay between the reference sensor and the loudspeaker (secondary source) offers the opportunity to electrically reintroduce the anti-noise at a position in the sound field where it will cause cancellation to the noise components correlated with the reference signal [3].

A feedforward system is the only configuration capable of cancelling broadband noise if the reference signal is highly correlated with the primary noise (coherence condition) and if there is enough time to generate the anti-noise before the primary noise reaches the secondary source (causality condition). These conditions are highly dependant on the spacing between the reference sensor and the secondary source. Also, to achieve high coherence the reference sensor must be a microphone, preferably close to the noise source, in order to reduce the amount of ambient noise captured by the microphone.

A narrowband noise does not need to meet such conditions, because its predictability allows to assume that the signal stays the same. So, it is easier to obtain a reference signal from narrowband noise than from broadband noise. If only narrowband noise is expected to be cancelled, a microphone

is not the best sensor. This is because of the acoustic feedback problem between the reference sensor and secondary source, generally observed in feedforward systems. Instead, a non-acoustic sensor, such as a tachometer, an accelerometer, or an optical sensor, should be used to synchronize with the noise source.

2.1.3 Feedback systems

Feedback control systems differ from feedforward systems in the manner in which the control signal is derived. Whereas feedforward systems rely on advanced information about the incoming disturbance to generate an appropriate anti-noise, feedback systems aim to attenuate the residual effects of the primary noise after it has passed [12]. Therefore, the adaptive feedback controller is only capable of cancelling narrowband noise, relying on its periodicity, because there is no way to have advanced information about the signal. The concept is illustrated in figure 2.2 for the duct noise problem. There, it can be seen that a feedback controller derives a control signal by filtering an error signal, not by filtering a reference signal as is done by a feedforward controller.

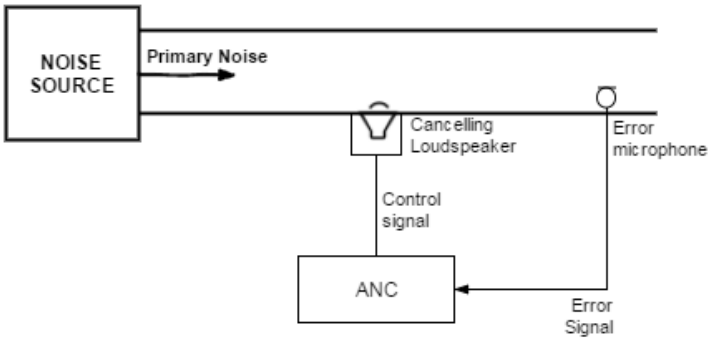


Figure 2.2: Single-channel feedback ANC system in a duct.

They are also implemented using a fixed controller however better attenuation for narrowband signals may be achieved with an adaptive controller [3]. The working principle of the feedback adaptive controller is to synthesize the primary noise by adding an estimate of the secondary noise to the error signal. Then this estimated primary noise is used as a reference signal to generate the secondary noise, proceeding as in an adaptive feedforward system. Thus the adaptive feedback system can be interpreted as a feedforward ANC using an estimated reference signal, whose future samples are a linear function of the previous samples.

Feedforward ANC is generally more robust than feedback ANC, particularly when the feedforward system has a reference input isolated from the secondary source. However, because the feedback system requires only one error microphone, it avoids the acoustic feedback problem inherent in a two microphone feedforward system [3].

2.2 Adaptive Controller

Adaptive filters play an important role in ANC systems as they are generally at the heart of an adaptive controller. This section will describe the general adaptive filter and explain its basic functioning, present the most important filter structures and will show the derivation to the most used adaptive algorithm and its modified versions, making known some of its properties.

2.2.1 Adaptive Filters

An adaptive filter is a system composed of a linear filter that has a transfer function controlled by variable parameters and a means to adjust those parameters according to an optimization algorithm. Because of the algorithms' complexity, almost all adaptive filters are digital filters. Contrary to digital filter with fixed coefficients, where it is necessary to assume time-invariance, know the filter specifications and the desired output signal a priori [25], with adaptive filters the time invariance restriction is removed in order to allow them to automatically adapt (self-optimize) to unknown changing environments and track time variations of the input signals.

Figure 2.3 depicts a general adaptive filter. Its inputs are the desired response (or primary input signal) $d(n)$, and the reference input signal, $x(n)$. The output signals are the output of a programmable digital filter driven by $x(n)$, $y(n)$, and the error signal $e(n)$, obtained from the difference between $d(n)$ and $y(n)$ at time n. The algorithm objective is to update the filter weights to find a filter to be applied to $x(n)$, in a way that the output $y(n)$ is as close as possible to $d(n)$. This will result in the error being progressively minimized on a sample-by-sample basis [3].

Adaptive filters are defined by four aspects [26]:

•the signals being processed by the filter;

•the structure that defines how the output signal of the filter is computed from its input signal;

•the coefficients within this structure that can be iteratively changed to alter the filter's input-output relationship;

•the adaptive algorithm that describes how the coefficients are adjusted from one time instant to the next.

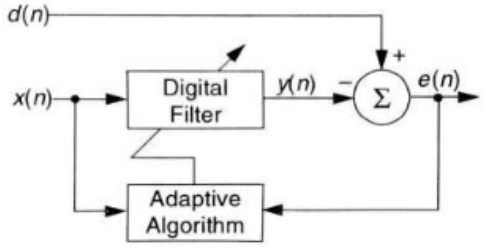


Figure 2.3: Block diagram of adaptive filter [3].

The linear filter may be realized using finite impulse response (FIR) or infinite impulse response (IIR). FIR filters incorporate only zeros and hence they are always stable and can provide a linear phase response. They are realized with a transversal structure or as lattice filters, a very useful structure for

adaptive filtering tasks. They are useful for applications which need to model an all-zeroes unknown system or as a simple and robust solution.

IIR filters can describe the response of a system more accurately or model an all-pole or pole-zero unknown system with a lot less taps and arithmetic operations than a FIR filter approximation for the same system. Thus they are used when computational complexity becomes an important issue. However, their feedback structure make their transfer function very sensitive to changes in the coefficients, making this class of filters unstable for certain applications and optimization procedures. They may be realized with various structures types, such as direct form I and II, their transposes, cascaded biquads [27]...

2.2.2 Adaptive Algorithms

Adaptive algorithms can take on a myriad of forms and are often derived as a form of optimization procedure that minimizes an error criterion (cost function) that is useful for the task at hand. Filter structure and algorithm are inter-related, with quantization errors, ease of implementation, computational complexity, among others, being the deciding factors [26].

There are two main approaches to the development of adaptive filter algorithms, the Stochastic Gradient Approach and Least Squares Estimation [28]. The Least Mean Squares (LMS) algorithm, one of the most commonly used algorithms in ANC, is a form of the stochastic gradient approach where the filter is only adapted based on the error at the current time.

The LMS algorithm uses a tapped delay line or transversal structure, i.e. a FIR filter, with L taps as seen in the diagram of Figure 2.4 where the output is computed with

$$y(n) = \sum_{l=0}^{L-1} w_l(n)x(n-l), \tag{2.1}$$

where $w_l(n)$ is lth coefficient of the adaptive filter.

It is useful to represent 2.1 in vector form

$$\begin{aligned} y(n) &= \mathbf{w}^T(n)\mathbf{x}(n), \\ \mathbf{w}(n) &= [w_0(n)\ w_1(n)\ \dots\ w_{L-1}(n)]^T \\ \mathbf{x}(n) &= [x(n)\ x(n-1)\ \dots\ x(n-L+1)]^T \end{aligned} \tag{2.2}$$

with $\mathbf{x}(n)$ as the tap-input vector at time instance n and $\mathbf{w}(n)$ as the tap-weight vector at time instance n.

Such systems are currently more popular than adaptive IIR filters because the FIR filter structure for any set of fixed coefficients guarantees the input-output stability [26], and adaptation of IIR filters can converge to a local minimum instead of a global minimum of the optimal filter coefficients space, therefore FIR algorithms are simpler. Another reason for IIR filters relatively narrow application stems from the insufficient research of adaptive IIR filter theory, since their analysis includes nonlinear systems of high order[27].

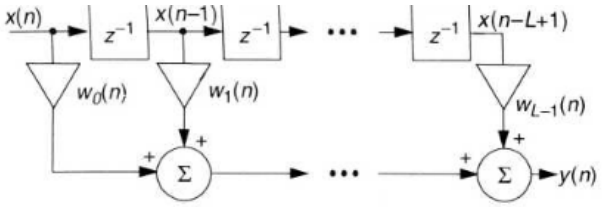


Figure 2.4: Block diagram of digital FIR (transversal) filter [3].

The Mean-Squared Error Cost Function

The most used criterion to find the optimal FIR filter coefficients is the minimum mean square error (MSE). The error signal $e(n)$ is defined according to Eq. 2.1 in the following manner

$$e(n) = d(n) - y(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n). \tag{2.3}$$

The MSE cost function is represented as

$$C(n) = E[e(n)^2], \tag{2.4}$$

where $C(n)$ is a cost function at time instant n and $E[.]$ denotes the expectation value operator. The MSE is useful for adaptive FIR filters because it is a quadratic function [3] and therefore

•it has a well-defined minimum with respect to the coefficients in $\mathbf{w}(n)$;

•the coefficient values obtained at this minimum are the ones that minimize the power in $e(n)$, indicating that $y(n)$ has approached $d(n)$;

•it is a smooth function of each of the coefficients in $\mathbf{w}(n)$, such that it is differentiable with respect to each of the coefficients in $\mathbf{w}(n)$ [26] .

To find the optimum coefficients that minimize the MSE, the gradient of the cost function with respect to the coefficients is equated to zero.

$$\nabla C(n) = -2E[\mathbf{x}(n)e(n)] = 0, \tag{2.5}$$

Continuing to develop Eq. 2.5 with Eq. 2.3, assuming that $e(n)$, $d(n)$ and $x(n)$ are stationary (the statistical properties of these signals do not change with time) and that the elements of the vector **w** are constant, yields

$$\mathbf{w}_*^T E[\mathbf{x}^T(n)\mathbf{x}(n)] = E[\mathbf{x}(n)d(n)], \tag{2.6}$$

where $\mathbf{w}_*$ denotes the optimum weight vector. The above equation can be rewritten as

$$\mathbf{R}\mathbf{w}_* = \mathbf{p}, \tag{2.7}$$

where **R** is the autocorrelation matrix of the reference input signal and **p** is the cross-correlation vector of the reference input signal and the desired response. The above equation is known as the Wiener-Hopf equations and provides a solution to the adaptive filtering problem in principle. However, in practical

applications with non-stationary noise, it is not possible to calculate the optimal filter coefficients from correlation properties of the incoming signals, since **R** and **p** are not available.

The Steepest Descent Procedure

Instead of solving 2.7, the optimum solution can be found by the well-established optimization procedure called *steepest descent*, which uses the gradient vector to gradually descend step by step to the minimum of the error function. It is not the faster method however it is the simplest. This procedure adjusts each parameter of the system according to

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \frac{\mu}{2}\Delta_w C, \tag{2.8}$$

where μ is the step size and $\Delta_w C$ is a vector of derivatives $\partial C(n)/\partial w_i(n)$. In other words, the ith parameter of the system is altered according to the derivative of the cost function with respect to the ith parameter.

This algorithm cannot be used in real-time applications because it requires exact knowledge of the MSE, when in practice only $d(n)$ and $x(n)$ are available. An approximate version of the method of steepest descent that depends on the signal values is used instead. This procedure is known as the LMS algorithm [26].

2.2.2.1 LMS Algorithm

The LMS algorithm was developed based on the Wiener filter, but the difference is that it does not assume any statistical knowledge of the signals [17].

Given that usually there is not enough information about the signals to calculate the autocorrelation and cross-correlation, Widrow used the instantaneous squared error, $e^2(n)$, to estimate the mean-square error given in equation 2.4. Therefore, the gradient estimate used by the LMS algorithm is simply the instantaneous gradient of a single squared error sample, i.e. $\partial E\{e^2(n)\}/\partial\mathbf{w}(n)$ is replaced by $\partial e^2(n)/\partial\mathbf{w}(n)$ [25]. The instantaneous gradient of a single squared error sample is

$$\frac{\partial e^2(n)}{\partial \mathbf{w}(n)} = 2\frac{\partial e(n)}{\partial \mathbf{w}(n)}e(n). \tag{2.9}$$

Using the gradient estimate in the steepest descent procedure Eq. 2.8, as described, gives the LMS algorithm expression

$$\begin{aligned}\mathbf{w}(n+1) &= \mathbf{w}(n) - \frac{\mu}{2}\frac{\partial e^2(n)}{\partial \mathbf{w}(n)} \\ &= \mathbf{w}(n) + \mu e(n)\mathbf{x}(n).\end{aligned} \tag{2.10}$$

where μ is the convergence factor (or step size) and $e(n)$ is the error at time instance n.

Important performance measures for adaptive algorithms are their stability, convergence time and, for algorithms using the minimum MSE criterion, misadjustment.

Stability is related to the algorithm ability to converge to the desired filter. In order for this algorithm

to be stable μ should verify

$$0 < \mu < \frac{2}{LP_x}, \tag{2.11}$$

where L is the adaptive filter length and P_x is the power of $x(n)$. One of the conclusions one can take from Eq. 2.11 (stability condition) is that small μ should be used for large-order filters, since μ is inversely proportional to L [3].

Convergence time is the time required for the algorithm to converge to the optimum least squares solution. It depends on the step size, since the larger the μ the faster the convergence. However it is a major concern for the LMS algorithm, since it is slow to converge for signals with a large spectral dynamic range, or equivalently a large eigenvalue spread[1] [28].

Misadjustment is the excess MSE over the minimum MSE. It is proportional to filter length, step size and input signal power. There is a tradeoff between convergence time and misadjustment. When μ increases, the convergence is faster but misadjustment increases; if μ decreases, the convergence is slower and misadjustment decreases [3].

Despite its limitations, the LMS algorithm is very popular and the most widely used learning algorithm for its simplicity, ease of computation, low memory requirements, because it does not require off-line gradient estimations or repetitions of data [17]. It has robust performance because it is model independent [28], meaning it has the ability to operate satisfactorily with ill-conditioned data (e.g. very noisy environment, change in signal and/or noise models).

Normalized LMS

As stated, stability and convergence time of the LMS algorithm is determined by the step size, which has a bound dependent on the power of the input data P_x. Therefore, these performance parameters vary with $x(n)$. To defeat this inconvenience one useful approach is to normalize μ with respect to the input signal power P_x, resulting in the algorithm called normalized LMS algorithm (NLMS) [28, 17].

The adaptation step size will vary with an estimate for the input signal power, $\hat{P}_x$. The most widely used way to calculate $\hat{P}_x$ for the NLMS is with a squared Euclidean norm of the tap-input vector $\mathbf{x}(n)$, before the increment step of the algorithm, as seen in Eq. 2.12 [3].

$$\hat{P}_x(n) = c + \|\mathbf{x}(n)\|^2 = c + \sum_{l=0}^{L-1} x^2(n-l). \tag{2.12}$$

with c as a small positive constant to avoid division by a very small P_x or zero. The NLMS algorithm is then defined as

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu e(n)\mathbf{x}(n)}{\hat{P}_x(n)}. \tag{2.13}$$

where the constant μ is the normalized step. This algorithm is faster than the ordinary LMS and it is convergent if the following condition on μ is satisfied.

$$0 < \mu < 2, \tag{2.14}$$

[1] ratio of the largest to smallest eigenvalue of the correlation matrix

Leaky LMS

When, for example, there is no input signal $x(n)$ or when using a noiseless sinusoid, the LMS algorithm may result in divergence of the adaptive weights. This happens because the weight update stalls, i.e. the weight increments are very small, and finite-precision effects can cause the unconstrained weights to grow without bound, resulting in overflow during the weight update process. One solution to the problem is based on adding a parameter to the update equation, referred to as leakage factor, which tends to bias each filter weight toward zero. The principle of this method is similar to adding white noise to the input signal, prior to the adaptive filter [3].

So, according to equation 2.10, this leaky LMS algorithm can be expressed as:

$$\mathbf{w}(n+1) = \lambda \mathbf{w}(n) + \mu e(n)\mathbf{x}(n). \tag{2.15}$$

where λ is the leakage factor with $0 < \lambda \leq 1$. The value of the leakage factor is in general determined by the designer on an experimental basis, as a compromise between robustness and loss of performance of the adaptive filter, due to the white noise addition. Usually a good starting point is a leakage factor slightly less than 1.

It has been shown that leaky LMS can be used to improve stability in a finite-precision implementation, mitigating the effects of non-persistent excitation, and reduce undesirable effects like stalling, bursting, etc [29].

2.2.3 Adaptive filtering applications

Adaptive filters have a wide range of applications which can be described in terms of more-general problem classes that describe the assumed relationship between $d(n)$ and $x(n)$. These are, generally, four basic classes: identification, inverse modelling, prediction, and interference cancelling. To get a better understanding of the ANC algorithms it is useful to present the system identification problem.

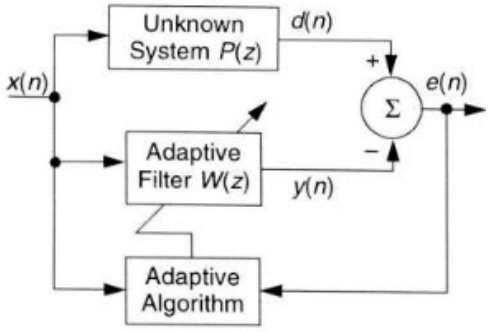


Figure 2.5: Adaptive system identification [3].

Adaptive system identification

System identification is an experimental approach to the modelling of a process or a plant. The basic idea is to measure the signals produced by the system and to use them to construct a model. Figure 2.5

shows the general problem of system identification, where $P(z)$ is an unknown system to be identified and $W(z)$ is an adaptive filter to estimate $P(z)$ [3].

It works by exciting both the unknown system $P(z)$ and the adaptive model $W(z)$ with the same input signal $x(n)$. The adaptive filter adjusts itself in order to minimize the error signal $e(n)$, computed with the difference between the physical system response $d(n)$ and adaptive model response $y(n)$. When $e(n)$ has been minimized, the adaptive model reproduces $P(z)$, implying that ideally $y(n) = d(n)$. In actual applications, however, there will normally be additive noise present at the adaptive filter input and so the filter structure will not exactly match that of the unknown system.

2.3 ANC Algorithms

With the knowledge acquired up until this point it is still not possible to implement an ANC system in an effective and robust way. This is because a real ANC system, realized on a DSP, has aspects that need to be taken into account in the filter-weight update algorithm. These are the presence of acoustical and electrical paths which modify the signals propagating in the system. Their transfer functions should be identified and their effects compensated by the controller. Therefore, this section will present ANC algorithms capable of compensating the path common to all system configurations, the secondary path.

It begins by describing the single-channel feedforward system as a system identification problem, with the existing electro-acoustical paths detailed. Then, one of the most widely used ANC algorithms, the filtered-x LMS algorithm (FxLMS), is explained. The secondary path modelling methods are presented next. Two examples of the FxLMS various forms, also included in this work, are introduced. They are the product of adapting the FxLMS to the mentioned estimation methods or to compensate for other effects.

The definitions are all made with the feedforward system as example. However, as stated, the feedback system can be interpreted as a feedforward system with a synthesized reference signal. Thus, the synthesis process will be described in more detail in this section.

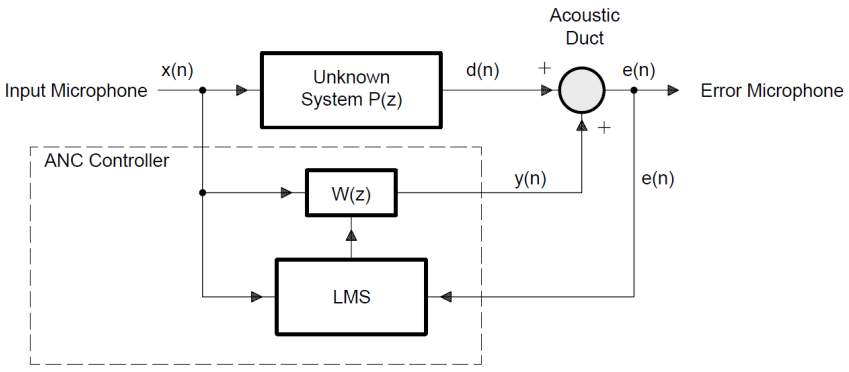


Figure 2.6: System Identification Approach to Feedforward ANC [4].

2.3.1 ANC as system identification

ANC systems can be described in a system identification framework as shown in Figure 2.6. The ideal ANC system uses an adaptive filter $W(z)$ to estimate an unknown plant $P(z)$. The unknown plant, also called primary path, consists of the acoustic response from the reference sensor to the error sensor. If the plant is dynamic, the adaptive algorithm then has the task of continuously tracking time variations of the plant dynamics.

It is convenient to make a correspondence of the ANC signals, mentioned in subsection 2.1.2, to the adaptive filter signals, thus bringing those to the context of active noise cancellation. The desired response [2] $d(n)$ corresponds to the primary noise as captured by the error microphone. Furthermore, the control signal of ANC coincides with the output of the adaptive filter $y(n)$, the reference signal to the reference input signal $x(n)$ and the error signal to $e(n)$.

An important difference between the system depicted in Figure 2.6 and the traditional system identification scheme (Figure 2.5) lies in how $e(n)$ is obtained. Instead of a subtractive junction, the summing junction used in Figure 2.6 represents the acoustic superposition occurring in the space from the cancelling loudspeaker to the error microphone, where the primary noise $d(n)$ is combined with the secondary noise, generated by $y(n)$. To force the destructive interference the control signal must be subject to a sign change before forwarding to the acoustical summing junction, thus generating the error signal. So, one can represent an ANC system block diagram with the subtractive junction, as in the adaptive filter convention, since the inversion will be implicit.

The block diagram of Figure 2.6 suggests that the control signal is directly combined with the primary noise and the resulting error signal is directly fed back to the control. However, in practice, that system is incomplete. The reason is that a real feedforward system, represented in Figure 2.7, has a number of electro-acoustic paths whose transfer functions must be included in the model.

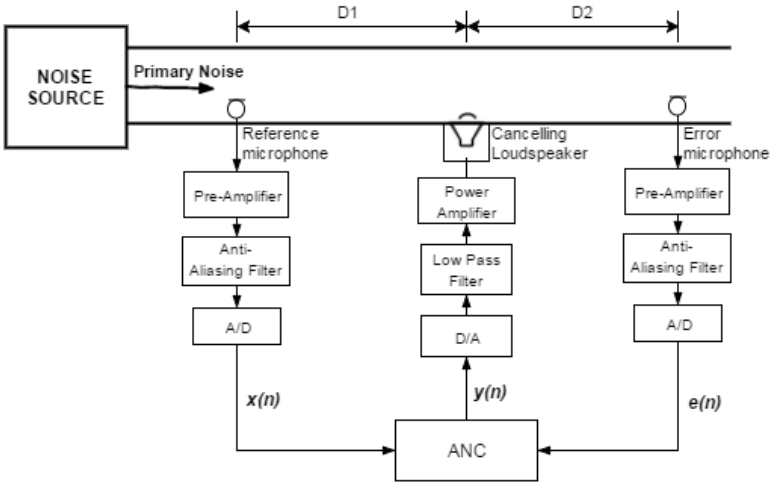


Figure 2.7: Real single-channel feedforward ANC system in a duct.

On these terms, the primary path description can be improved. The electrical reference signal $x(n)$ is

[2] It should be noted that "desired" is a generic term used with adaptive filters to denote a training signal. In the context of ANC this signal is undesired [17].

obtained from the acoustic pressure picked by a microphone but one can interpret the conversion backwards and see it as driving the plant and the adaptive filter. Therefore the primary path transfer function $P(z)$ consists of the A/D (analogue-to-digital) converter (ADC), the anti-aliasing filter, the pre-amplifier, the reference microphone and the acoustic path from reference microphone to error microphone transfer functions cascaded. Note that the ADC and anti-aliasing filter frequency response appear inverted in the expression of the path. If the A/D and D/A are symmetric then their responses cancel out and the result is only the acoustic path.

Secondary path

As mentioned before, the electrical error signal is sensed from the residual acoustic noise using an error microphone. This pressure wave results from the acoustic superposition of the primary noise and the anti-noise, generated from a loudspeaker driven by the control signal $y(n)$. Therefore, it is necessary to compensate for the secondary-path, also called cancellation path, transfer function $S(z)$ from $y(n)$ to $e(n)$. This path includes the D/A (digital-to-analog) converter (DAC), reconstruction filter, power amplifier, loudspeaker, acoustic path from loudspeaker to error microphone, error microphone, preamplifier, antialiasing filter, and ADC [4].

Observing Figure 2.7 one can point out that the primary path and secondary path share the path from the cancelling loudspeaker to the error microphone. However, as demonstrated in [3], the primary path transfer function and secondary path can be considered as separate entities and the feedforward ANC system can be represented through the system identification viewpoint as in Figure 2.8.

To compensate the secondary path effects the transfer function must be modelled, also using a system identification approach, and the estimate must be included in the adaptive algorithm as seen in the following section.

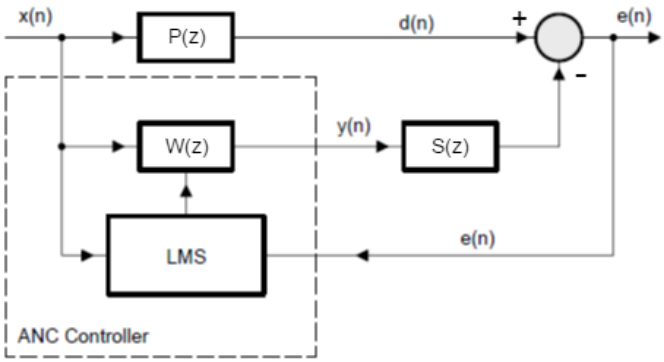


Figure 2.8: Block diagram of ANC system including the secondary path [4].

Acoustic feedback

The use of a reference microphone introduces the possibility of undesired feedback from the secondary source to the reference input microphone. Thus the overall system forms a closed loop, complicating the problem of adaptation, or rendering the system unstable. This path is the acoustic feedback path with transfer function $A(z)$. There are many more of these path in multiple-channel systems so

this is a major problem for applications needing that type of system. It should be noted that an acoustic feedback path is not formed between the error microphone and the secondary source. This is because the error signal is used by the adaptive algorithm and the secondary noise is generated by filtering the reference signal, meaning they are electrically isolated from each other during operation and no loop is formed.

One can avoid acoustic feedback by using directional reference microphones pointing away from the secondary source or using non-acoustic reference sensors, as they are insensitive to the cancelling sound [4]. It is also advantageous to locate the reference microphone far upstream from the cancelling loudspeaker in order to have a reduced acoustic feedback. Another approach is to use an algorithm which compensates for the path's effects in the same way as with the secondary path.

2.3.2 Filtered-x LMS Algorithm

As represented in Figure 2.8, there is a transfer function after the adaptive filter. In the ideal case, the optimal filter coefficients are the same as in the primary path transfer function, but with the secondary path the optimal filter becomes

$$W_*(z) = \frac{P(z)}{S(z)}. \tag{2.16}$$

The performance of an ANC system is tied to the secondary path transfer function $S(z)$ and, to ensure convergence, the secondary path needs to be compensated.

Morgan [30] suggested two ways to modify the LMS algorithm to compensate the secondary path:

•to place an inverse filter, $1/S(z)$, in series with $S(z)$ to remove its effect;

•to place an identical filter in the reference signal path to the weight update of the LMS algorithm, realizing the so-called filtered-X (FxLMS) algorithm.

Since an inverse does not necessarily exist for $S(z)$ and placing it in the secondary path would need a sharp filter with long delay [23], the FxLMS algorithm is generally the most effective approach.

Derivation of FxLMS algorithm

Figure 2.9 shows an active noise control system scheme with the secondary path located at the output of the adaptive filter and its estimate placed at the input of the LMS algorithm. The signals can be expressed as:

$$\begin{aligned} e(n) &= d(n) - y'(n) \\ &= d(n) - s(n) * y(n) \\ &= d(n) - s(n) * [\mathbf{w}^T(n)\mathbf{x}(n)], \end{aligned} \tag{2.17}$$

where $s(n)$ is the impulse response of secondary path $S(z)$ at time n, $*$ denotes linear convolution, $y'(n)$ is the anti-noise,

$$\mathbf{w}(n) = [w_0(n)\ w_1(n)\ \ldots\ w_{L-1}(n)]^T \tag{2.18}$$

is the coefficient vector of $W(z)$ at time instance n,

$$\mathbf{x}(n) = [x(n)\ x(n-1)\ \ldots\ x(n-L+1)]^T \tag{2.19}$$

is the signal vector at time instance n, and L is the order of filter $W(z)$.

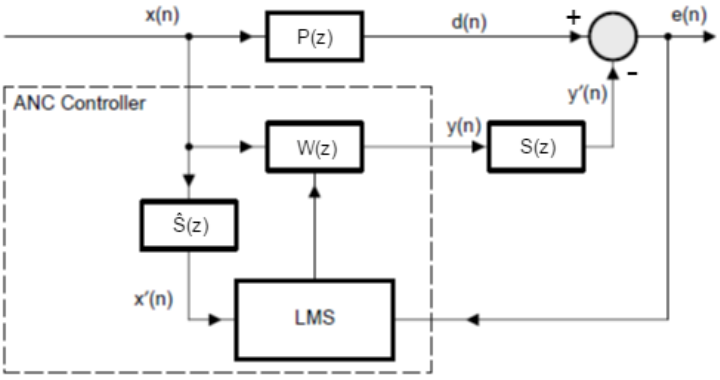


Figure 2.9: Block diagram of ANC system using the FxLMS algorithm [4].

Substituting Equation 2.17 in the LMS expression (Equation 2.10) and solving the gradient gives the expression for the FxLMS algorithm [3]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}'(n), \tag{2.20}$$

where

$$\mathbf{x}'(n) = [x'(n)\ x'(n-1)\ \ldots\ x'(n-L+1)]^T \tag{2.21}$$

and

$$x'(n) = s(n) * x(n). \tag{2.22}$$

In practical ANC applications $S(z)$ is unknown and estimate of it $\hat{S}(z)$ is used. So, $\mathbf{x}'(n)$ is evaluated as:

$$\mathbf{x}'(n) = \hat{\mathbf{s}}(n) * \mathbf{x}(n). \tag{2.23}$$

where $\hat{\mathbf{s}}(n)$ is the coefficient vector of the estimate of the secondary-path filter impulse response.

Leaky FxLMS Algorithm

The leaky FxLMS algorithm solves another problem introduced by the direct application of the FxLMS, besides the problems presented in section 2.2.2.1. This problem is the occurrence of non-linear distortion on the secondary source from overdriving it with the control signal due to high noise levels associated with low frequency resonances. The use of the leakage factor limits the output power of the secondary source to avoid the nonlinear distortion. It has been shown that the introduction of a leakage factor has a considerable stabilizing effect on the adaptive algorithm, especially when very large source strengths are used [3].

Feedback FxLMS algorithm

In a single-channel feedback ANC system the reference input microphone is not used, therefore a reference signal is not available. An advantage of this is that it avoids the acoustic feedback problem inherent in the two-microphone feedforward systems, that were discussed previously.

From Equation 2.17 one can conclude that the primary noise signal $d(n)$ can be expressed in the z-domain as:

$$D(z) = E(z) + S(z)Y(z), \tag{2.24}$$

where $E(z)$ is obtained from the error microphone and $Y(z)$ is the output of the adaptive filter. If the transfer function $S(z)$ of the secondary path is modelled by $\hat{S}(z)$, $D(z)$ can be approximated as:

$$\hat{D}(z) = E(z) + \hat{Y}'(z). \tag{2.25}$$

where

$$\hat{Y}'(z) = \hat{S}(z)Y(z) \tag{2.26}$$

is an estimate of the anti-noise. In essence, an anti-noise estimate $\hat{y}'(n)$ and $e(n)$ are added to electrically undo the destructive interference, thus synthesizing the reference signal $x(n)$ as an estimate of the primary noise $\hat{d}(n)$ [3].

Then, the feedback FxLMS algorithm takes the form depicted in Figure 2.10. What effectively happens in the algorithm is the regeneration of the periodic signal and then trying to cancel the phase difference between the regenerated reference signal and the primary noise.

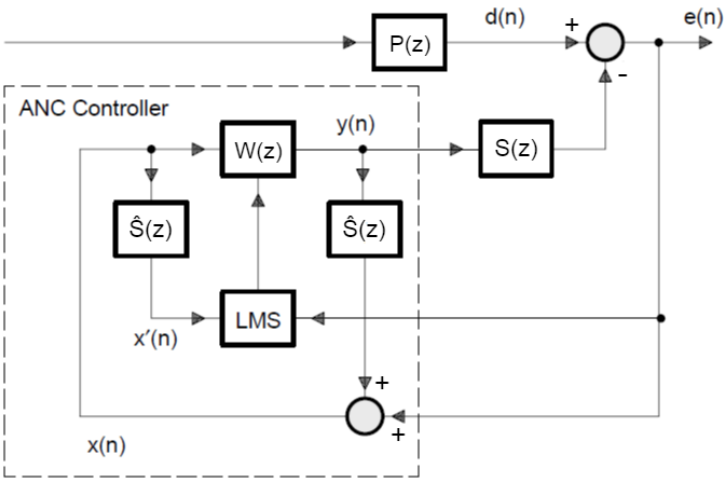


Figure 2.10: Block Diagram of the Feedback ANC System [4].

2.3.3 Secondary path modelling

The FxLMS algorithm needs an estimate of the secondary path transfer function $\hat{S}(z)$ to obtain the filtered-x signal $x'(n)$, because the secondary path transfer function $S(z)$ is unknown. Its important for $\hat{S}(z)$ to be fast and reasonably accurate in order to ensure adequate performance, stability and convergence speed of the system.

The model can be constructed with the system identification approach of adaptive filters. However, the secondary path transfer function is time varying due to effects such as aging of the loudspeaker,changes in temperature, and air flow in the secondary path [4]. So, the method to obtain the

estimate $\hat{S}(z)$ depends on the degree and speed of variation. The variations can be classified into three types [12]:

•if the variation of $S(z)$ is within a small range, the model may be obtained off-line during a training stage and be used permanently in the control filter weight-update algorithm;

•if the change is slow or irregular, but extensive over a long period of time, the control system must implement the secondary path modelling on-line. This means the model identification is occasionally or continuously done at the same time as the weight-update of the control filter;

•if $S(z)$ varies rapidly by a large amount the FxLMS algorithm might not be suitable for the application if it cannot determine the correct $\hat{S}(z)$ in a short time.

Off-line modelling approach

As stated, if $S(z)$ is time-invariant but unknown, off-line modelling can be used to construct the model $\hat{S}(z)$ during an initial training stage. At the end of the training interval, the estimated model $\hat{S}(z)$ is fixed and used for ANC operation.

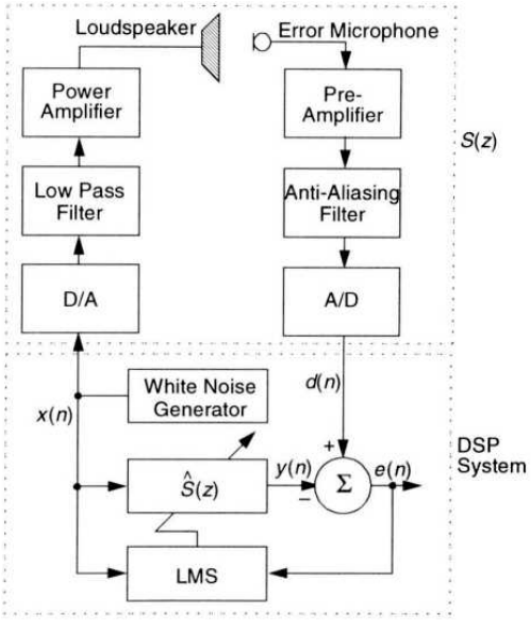


Figure 2.11: Experimental set up of secondary path modelling [3].

The experimental set up for direct off-line system modelling is shown in Figure 2.11. An internal white noise signal is internally generated by the DSP system, depicted as the random noise generator, and sent to the speaker. White noise is used as a training signal because it has a constant spectral density at all frequencies. This makes the LMS algorithm converge fast. The microphone picks up the noise and its output is subtracted from the filter output to form the error signal. A LMS algorithm updates the filter coefficients to minimize the error signal. After convergence, the modelling filter $\hat{S}(z)$ is an estimate of the unknown secondary path transfer function, $S(z)$.

Although the off-line modelling technique results in a functional ANC system, slight changes in the environment during system operation may lead the adaptive algorithm to instability. A phase difference between $S(z)$ and $\hat{S}(z)$ in excess of 90° will force this result. Even though the system remains stable,

convergence will slow appreciably as the phase difference approaches $90°$ [3]. A way to counter this is to block noise control for frequencies whose phase error usually exceeds $90°$, or use the on-line modelling approach for slow and continuous variations.

On-line modelling approach

Adaptive on-line modelling of the secondary path should be used when $S(z)$ continuously varies in real time. $S(z)$ can be either occasionally or continuously estimated and the most recent estimate $\hat{S}(z)$ used in the weight update of the controller. Provided that $S(z)$ changes slowly, the controller adaptation and secondary-path estimation functions can be considered separately.

The additive random noise technique [3] is a popular method of on-line secondary path modelling. In summary, it adds to the control signal an internally generated white noise to drive the secondary source. Thus, the error signal will have a component tied to the primary noise and a component tied to the white noise. The algorithm will try to use each component separately to weight update the control filter for the first component, and the secondary path model for the second component. However to do that is not trivial and may degrade the performance of each process. While this may seem to be a somewhat counter-productive exercise, injecting an additional disturbance into a system targeted for active control, the level of the modelling disturbance can be very low (say, 30 dB below the unwanted primary disturbance) and still provide a model of suitable accuracy over a relatively long period of time [12].

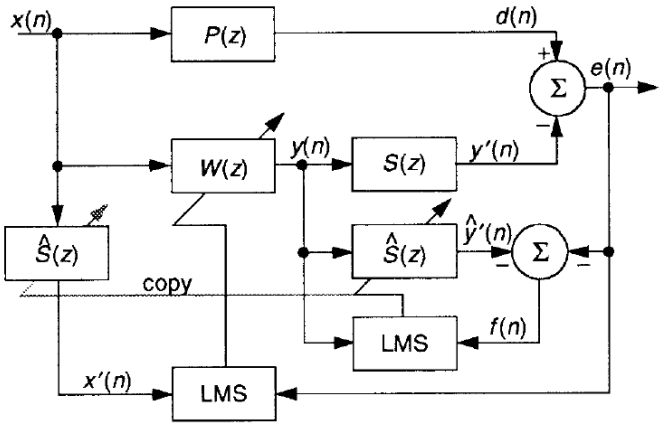


Figure 2.12: Block diagram of an online secondary-path modelling [3].

The overall modelling algorithm (OMA) used in an algorithm in this work, is another popular method of on-line secondary path modelling. Instead of using a generated training system it uses the existing signals in the system to estimate the secondary path transfer function. To explain it, it is useful to first present a secondary path modelling technique proposed in [17], represented in Figure 2.12.

The optimal solution of $\hat{S}z$ for this system is [3]

$$\hat{S}_*(z) = S(z) - \frac{P(z)}{W(z)}. \tag{2.27}$$

This equation shows that the estimate $\hat{S}(z)$ is biased by $P(z)/W(z)$. It is useful to try to eliminate the

bias because the adaptive filter $\hat{S}(z)$ can correctly identify $S(z)$ only if $P(z) = 0$ [or equivalently, $d(n)$ = 0]. That is exactly the objective of the OMA, depicted in Figure 2.13. It tries to model $P(z)$ with the adaptive filter $\hat{P}(z)$ to filter the reference signal with $\hat{P}(z)$. The output is then used to cancel the primary noise thus cancelling the effects of the biasing term.

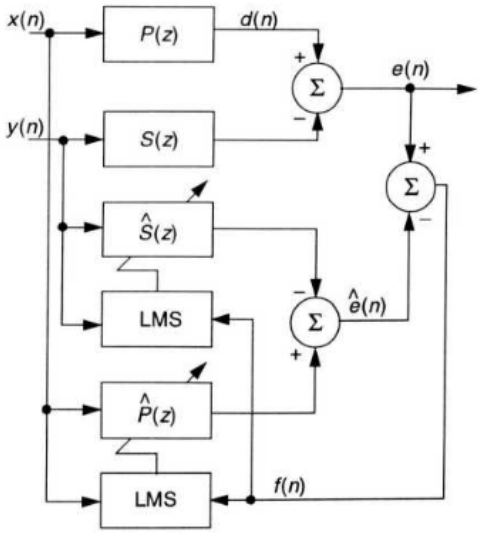


Figure 2.13: Overall on-line modelling algorithm [3].

Comparing both methods, the first on-line modelling approach, which involves injecting low-level random noise into the control signal, is superior in terms of convergence speed of both the control filter and secondary path modelling filter, for speed of response to modifications in the primary noise and the secondary path, for independence between the primary noise attenuation and the on-line secondary path identification, and for minimal computational complexity. However, the second approach does not add any additional noise into the system and usually requires less memory to implement [12].

2.3.4 Band limited Modified FxLMS algorithm

A way to circumvent instability stemming from excessive phase error between $S(z)$ and $\hat{S}(z)$, after using the off-line method for secondary path modelling during the training stage, is to block noise control for the frequencies where this phenomenon occurs more frequently, as mentioned. This ”undesired band” can be identified by doing a series of off-line secondary path modelling trials, computing the maximum phase difference between the estimates and analysing which frequencies the phase error is superior to $90°$.

A way to impede the noise control is to filter out the undesired frequencies from the anti-noise, by putting a band limiting filter before the secondary source. However this approach would need a sharp filter with long delay, which is undesirable in ANC systems, since the controller would try to invert this filter [23]. Instead, a band limited modified FxLMS (MFxLMS) algorithm can be used. It is interesting to use this algorithm in a feedback system to compare the differences in performance between the feedback FxLMS and feedback band limited MFxLMS. The band limited feedback MFxLMS (BMFxLMS) is illustrated in Figure 2.14.

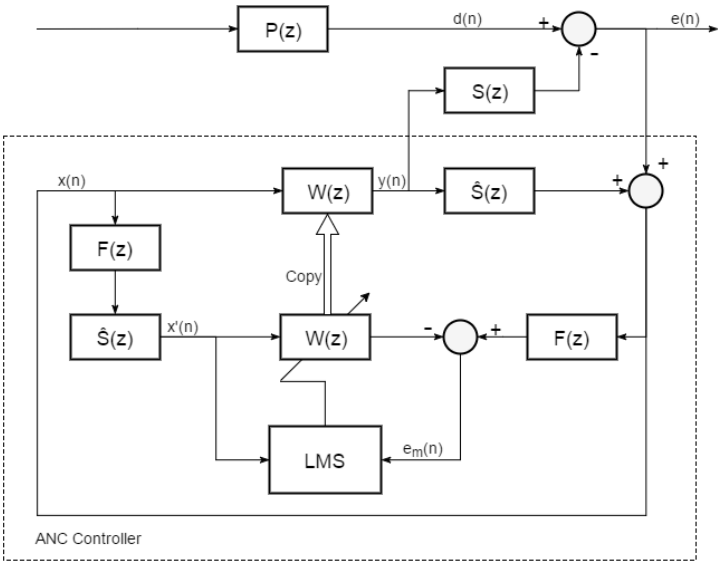


Figure 2.14: Block diagram of the Bandlimited feedback MFxLMS algorithm.

Another way could be by limiting the band of interest by filtering the primary noise $d(n)$, which corresponds to filtering the anti-noise. However, obviously, an acoustic signal cannot be electrically filtered so, alternatively, one can suppress the influence of the primary noise on the system by band limiting all the signals generated or derived from $d(n)$. The reference signal $x(n)$ can be filtered but, for a controller using the FxLMS algorithm, the error signal is acoustically obtained from the primary noise and anti-noise superposition. Thus, it is also difficult to filter the component of the error tied to the primary noise. That is where the MFxLMS algorithm turns out to be useful.

The MFxLMS algorithm predicts the error that would have been obtained if the current controller coefficients had been fixed over all time. The modified error $e_m(n)$ is obtained by using an internal model of the secondary path $\hat{S}(z)$ to derive an estimate of the primary noise $\hat{d}(n)$, which is then subtracted to the reference signal filtered by $\hat{S}(z)$ and the adaptive controller $W(z)$ [31]. This way the adaptation is done with an electrical model of the ANC system. Thus, all the necessary signals are electrically available and what is left is to band limit them and adapt the control filter using them.

This algorithm is also possible to realize for feedback systems. All it is needed is to use $\hat{d}(n)$ as a reference signal, as aforementioned, and filter both by the band limiting filter $F(z)$ before the weight update process of the control filter. Note that by filtering the reference and the desired signal the optimum filter is not changed. However the filter will not adapt on undesirable frequencies. This should be used in conjunction with the leaky LMS to make sure the filter goes to zero at these frequencies [23].

2.3.5 Mirrored MFxLMS algorithm [24]

As stated, the problem of variation of the secondary path can be solved with the on-line methods presented in the previous section. However, most of these algorithms become unstable after sudden changes, since they are only suitable to deal with slow changes. The mirrored MFxLMS (MMFxLMS) algorithm however, is stable even with incorrect secondary path models and deals very well with sudden changes.

It is a variation of the MFxLMS algorithm, with the OMA embedded in the algorithm for on-line modelling of the primary and secondary paths. However, instead of estimating the primary noise by adding an anti-noise estimate to the error signal, like the normal MFxLMS, it passes the reference signal through the primary path estimate $\hat{P}(z)$ obtaining the primary noise estimate $\hat{d}(n)$. The mirrored term comes from the fact that after obtaining the primary and secondary paths models one can construct a mirror copy of the acoustic paths, and then use the MMFxLMS.

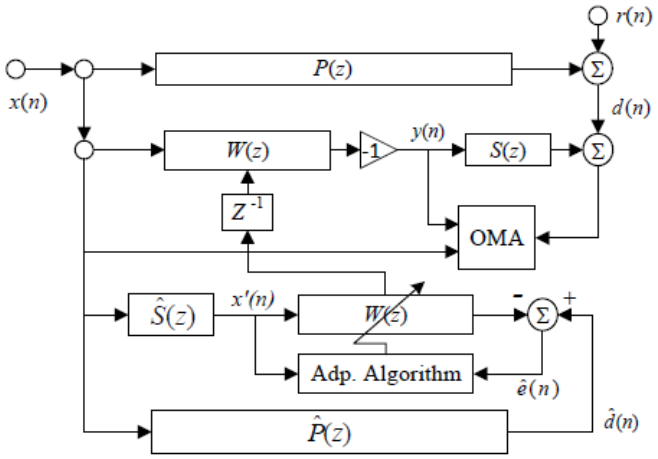


Figure 2.15: Block diagram of the MMFxLMS algorithm [24].

2.4 System Considerations

There are other limiting aspects of the system that limit the performance of the ANC system, which cannot be compensated by the controller. If they are not optimized the system may fail and therefore it is necessary to apply appropriate design techniques to assist the digital adaptive control.

This section will discuss the physical factors that limit the performance of this system, such as coherence, causality, sampling rate and filter length. The properties of acoustic spaces are not studied since they are more important in multiple-channel systems. Regardless, the use of a commercial headset for the system implementation has this problem, in principle, dealt with.

2.4.1 Coherence

Coherence describes all properties of the correlation between physical quantities of a single wave, or between several waves. Specifically, two wave sources are perfectly coherent if they have a constant relative phase and the same frequency. It is an ideal property of waves that enables stationary interference.

The coherence function [32] is a measure of the degree of linear dependence between two signals, as a function of frequency. It is calculated from the two auto-spectra (or power spectra) of signals $d(n)$ and $x(n)$ and their cross-spectrum as follows:

$$|\gamma_{dx}(\omega)|^2 = \frac{|S_{dx}(\omega)|^2}{S_{dd}(\omega)S_{xx}(\omega)}. \tag{2.28}$$

The cross and auto power spectra are the Fourier transform of the cross and autocorrelation functions. The coherence function satisfies the constraint

$$0 \leq |\gamma_{dx}(\omega)|^2 \leq 1 \tag{2.29}$$

at all frequencies, with 1 indicating a high degree of linear dependence between the two signals $d(n)$ and $x(n)$.

The performance of ANC can be determined by frequency-domain analysis of the error signal $e(n)$. The spectral representation of $e(n)$ is given by [3]

$$S_{ee}(\omega) = [1 - |\gamma_{dx}(\omega)|^2] Sdd(w), \tag{2.30}$$

This equation indicates that the performance of the ANC system, measured in terms of the spectral content of the residual signal, is dependent on the coherence. In order to realize a small residual error, it is necessary to have very high coherence [$C_{dx}(\omega) = 1$] at frequencies for which there is significant disturbance energy [1].

The coherence of two microphone signals will be less than unity in high-frequency diffuse fields, when sound field energy has no direction and sound pressure is the same everywhere in the acoustic space, or if the microphone signals are contaminated by electrical noise, unsteady air flow, cable vibration [12] or ambient noise unrelated with the unwanted noise. Note that diffuse sound fields is an approximation that is useful in many acoustics applications but that needs to be used carefully in ANC.

2.4.2 Causality

Causality is something one should be concerned when using feedforward systems. In these systems the reference signal provides information in advance about the primary noise before it reaches the cancelling loudspeaker, which is a necessary requirement for a causal controller. If causality is not possible, the system can effectively control only narrowband or periodic noise [3].

Figure 2.7 shows acoustical paths important to explain the conditions to guarantee causality. The acoustic delays suffered by the noises as they travel through the duct are proportional to the distances of the paths Dj:

$$\delta_{Aj} = \frac{Dj}{c_0}, \qquad j \in [1, 2] \tag{2.31}$$

where δ_{Aj} is the acoustic delay through path Dj and c0 is the speed of sound in air, which is 343 ms^{-1} at 20° C. The acoustic delay δ_{A2} is common for both primary noise and cancelling signal. Another important delay one should be aware is the electrical delay, expressed as:

$$\delta_E = \delta_W + \delta_T, \tag{2.32}$$

where δ_W is the group delay of the digital filter $W(z)$ and δ_T is the total system delay in the anti-aliasing filter, reconstruction filter, ADC, DAC, pre-amplifiers, power amplifiers, loudspeaker, microphone plus the

processing time (one sampling period, T_s).

In order for the system to be causal a couple of conditions must be guaranteed. The acoustic paths $D1$ and $D2$ must be dimensioned obeying

$$\delta_{A1} \geq \delta_{A2}. \tag{2.33}$$

in order to be possible to compensate for the inherent delay caused by $S(z)$, since the acoustic delay consists of a big part of that delay and the primary path must contain a delay of at least equal length [1]. The distance $D2$ may be very small as long as the error microphone does not capture the movement of the loudspeaker. It also must be ensured that

$$\delta_{A1} \geq \delta_E, \tag{2.34}$$

because the adaptive filter necessarily has a causal response. This last condition is called causality constraint, and it specifies the minimum length of a system for which a broadband random noise can be effectively cancelled.

Hence, to control broadband noise, it will be advantageous to locate the reference microphone far upstream from the cancelling loudspeaker in order to have a longer acoustic delay. Another way to assure causality is reducing the electrical delay by using a high sampling rate, specially in applications with space limitations [33]. Also, the loudspeaker should be selected carefully as its delay has a great influence on causality, especially at low frequencies [4].

2.4.3 Sampling rate and Filter Length

As stated, the controller must complete the entire signal processing task before the primary noise arrives at the loudspeaker. Real-time digital signal processing requires that the processing time t_p be less than the sampling period T_s. That is:

$$t_p < T_s = \frac{1}{F_s}, \tag{2.35}$$

where F_s is the sampling rate, which must be held high enough to satisfy the Nyquist criterion. That is:

$$F_s \geq 2f_M, \tag{2.36}$$

where f_M is the highest frequency of interest, approximately 500 Hz for most practical ANC applications.

However the choice of sampling rate is more complicated than choosing the Nyquist rate of the band of interest. The reason is the coder-decoder (CODEC)[3] having sampling rate limitations and since there are advantages and disadvantages to high sample rates comparing to low sampling rates.

As stated before, high sample rate is desirable because it would reduce the delay between the reference noise and anti-noise in a feedforward system, thus improving the conditions to meet the causality condition. Even though there is no causality condition for feedback systems delay is of importance

[3]The device containing the ADC, DAC, reconstruction filter and anti-aliasing filter

anyway, because excessive delays would limit the achievable performance and stability margin in a feedback system. This is especially important for ANC applications that generally involve relatively large bandwidth [33].

However, there are several points that may prohibit the use of high sample rates. First, low processing time would not allow for filters with a large number of coefficients. With not enough time to compute the filtering, causality issues would appear. Also, with a low bandwidth of interest, the effective control bandwidth of the system will be a small portion of the total frequency span. This results in the need for very high order FIR filters to ensure adequate accuracy in the magnitude and phase response. Lastly, numerical problems may arise so that a filter may easily become unstable [33].

So, it would be useful to process the data at a rate as low as the Nyquist frequency to enjoy the high processing time, lower filter order and still sample at the highest rate possible for the low delay. The multi-rate signal processing technique solves such problems as it combines the advantages of low delay, from high sampling frequency, with the FIR filter characteristics and performance, from low frequency. More specifically the reference and error signals are oversampled at the CODEC and the control computations are performed at a lower rate. A small introduction to multirate signal processing is made in Appendix A.

The length of the noise control filter depends upon the acoustics of the duct since the required length is reduced by the addition of passive damping material [8].

A trade-off must be considered when trying to meet the filter order condition. The reason is that, when cancelling broadband noise it is necessary that the filters be long enough to account for the distances of the paths. This is because the distance is related to the acoustic delay and, to model long delays, the appropriate number of samples to capture the impulse response. However, the filter order also depends upon the sampling rate. If, for example, a high sampling rate is used to reduce the delay even more the number of coefficients will be even bigger. Then, the processing time might not be enough to process the signals creating causality issues anyway. Thus, care must be had when dimensioning the feedforward system for broadband noises. For periodic signals such as sine waves, this constraint no longer applies, because only adjustments to phase over one cycle of the sine wave are required.

Another limitation imposed on the system is that the direct modelling filter must be sufficiently long to ensure adequate accuracy in the magnitude and phase response of the filter at the lowest frequency of interest.

2.5 ANC headset system

Active headsets enhance the performance of conventional passive earmuffs at low frequencies (usually below 500 Hz). The active system cancels low-frequency noises and the ear shell (passive) system attenuates high-frequency noises. Active headsets can be interpreted as a problem of noise propagating in ducts, since plane waves in ducts and the sound field in a hearing protector are both one-dimensional problems, thus enabling good results to be achieved with a single-channel control system [12]. Another reason for the good results is the size of the ear shell combined with the ear duct being about the same dimensions as the diameter of the cancellation volume near the error microphone. For example, a noise

of 500Hz corresponds to a radius of $\lambda/10 \simeq 7$ cm.

There are two main types of control system for active headsets; feedback and feedforward. In addition, feedback types may be further classified into analogue(fixed) and digital(adaptive), however analogue control, although the most widespread commercial headset controller, will not be focused in this work. Adaptive feedback headset systems are rather simple since the only design constraint is the choice and positioning of the loudspeaker and error microphone to avoid stability issues. As mentioned, it can only cancel narrowband noise but they don't have the problems inherent to feedforward system and therefore are the most used configuration.

Due to the advantages of feedforward systems such as better stability and robustness (without acoustic feedback), attenuation, bandwidth and ability to cancel broadband noise, it is desirable to implement the headset with such configuration. However, design of active feedforward headsets is more complicated due to considerations one should have with them. Though, the acoustic feedback, a critical problem in other feedforward applications, is virtually negligible [33] in headset applications. That is because the loudspeaker is directionally pointed to the ear and the anti-noise is blocked by the ear-shell, thus not creating a closed loop between the loudspeaker and the reference microphone.

There should be special care with the causality condition, because the available space is very limited, so the reference microphone will be very close to the error microphone. This means that the controller must act very fast or else causality will not be maintained and reduction of random noise will be limited. The noise source position referent to the sensors is specially important [34] since it may cause causality problems.

Maintaining coherence is another important problem because the reference microphone is located outside generally attached to the shell. Thus its measurements are heavily corrupted by wind [14] and the ambient noise level will be high compared to the primary noise. To counter these effects it is recommended, whenever possible, to obtain a synchronisation signal from a non-acoustic sensor. The system will be a lot more simplified since there will be no problems of coherence, however these kinds of systems only control narrowband noise.

Although it has been omitted, the used materials for passive cancellation, the shape of the shell and the tightness of the seal of the ear cup are also very important in the performance of a active headset.

Chapter 3

State-of-the-art

By reading the previous chapter, two books [3, 12] are constantly referenced. These are two of the best known books on ANC, the third being [8]. Good tutorial reviews are [1, 9] and Hansen also made a good introduction manual on active control of vibration and sound [35]. They give an overview including concepts such as broadband and narrowband feedforward ANC systems, adaptive feedback ANC systems, multiple-channel ANC, the FxLMS algorithm.

Other advancements of ANC systems mentioned on those books, have been researched. Hybrid ANC systems [20, 21, 22] are an example which employ both feedforward and feeback (adaptive or fixed controllers) configurations, to improve performance of broadband and narrowband cancellation, audio integration techniques[36, 37], to be possible to listen to music or other desired signals with the secondary source.

New techniques such as virtual sensing [38, 39], to form quiet zones away from error microphones, and new algorithms such as the ones based on the recursive least squares algorithms [40] and non-linear FxLMS based on non-linear filters such as multilayer artificial neural networks [41], have been developed and applied. For secondary path estimation there have been advancements in on-line modelling with improvements on the additive noise method [42, 43] and on the OMA [24]. Fujii in [44] proposes the simultaneous equations method, where the secondary path is estimated using two equations for two values of the controller filter and does not require additive noise. In [45] Fujii modifies the simultaneous equations method.

Directly regarding ANC headphones there are a number of patents [46, 47, 48] (on analog fixed feedback) and a number of papers. Some papers focus on audio integration on active headsets [49, 50]. The hybrid techniques mentioned were implemented in active headset systems. In [51] an ANC headset is developed to study the influence of error microphone position in the ear cup and to compare with commercial headphones. Headsets with different designs and controllers have been implemented, such as an in-ear noise cancelling headphones with an application-specific integrated circuit (ASIC) controller[52], a motorcycle helmet with ANC and audio integration [53] or an headset with a controller implemented with a micro-controller instead of a DSP has been investigated in [54]. There is a comparison of commercial headsets in [55].

Chapter 4

System Architecture

In this chapter the whole system will be described, with the hardware and software developed for the project having the design choices justified. The hardware developed was a analog circuit used to power and polarize the microphones, the signal conditioning circuit and the loudspeaker driver. The software implemented was the filtering function, the multi-rate signal converter (including the decimator and interpolator) and the algorithms described in chapter 2.

4.1 Hardware

The ANC system hardware is composed of on-ear headphones, housing a pair of microphones and speakers, a signal conditioning and speaker driving circuit, a DSP development board and a reference microphone with its pre-amplifier. This last one is the TMS320C6713 DSP starter kit (DSK) from Spectrum Digital and it has a DSK_AUDIO4 daughtercard, from Educational DSP, LLC, installed in the DSK's peripheral expansion slot.

4.1.1 C6713 DSK

The TMS320C6713, with its board layout represented in Figure 4.1, DSK is a standalone development platform that enables users to evaluate and develop applications for the TI C67xx DSP family. It also serves as a hardware reference design for the Texas Instruments (TI) TMS320C6713 DSP. This is a 32 bit floating point processor (also able to operate with fixed point arithmetic) based on a Very Large Instruction Word architecture, making it recommended for multichannel and multifunction applications. Operating at 225 MHz, the C6713 delivers up to 1350 million floating-point operations per second (MFLOPS), 1800 million instructions per second (MIPS), and with two fixed-/floating-point multipliers up to 450 million multiply-accumulate operations per second (MMACS). It has 6 dedicated arithmetic logic units (ALU) and 264 kB of internal memory [56].

The DSP has two possible ways to interface with analog audio signals; using the on-board TI CODEC, the TLV320AIC23B (AIC23), or using a daughtercard, connected to the expansion port of the board, with additional CODECs, the DSK_AUDIO4 daughtercard. The AIC23B has four 3.5mm au-

dio jacks available: a mono microphone input(biased), a stereo line input, a stereo line output, and headphone output (power amplifier included). The DSK_AUDIO4 daughtercard shares the arrangement of line level terminals with the on-board audio connectors, as seen in subsection 4.1.2. Both boards have these terminals AC coupled, meaning that there are DC_block capacitors following the connectors. To meet the objectives the DSK_AUDIO4 daughtercard was used for its increased number of terminals, thus enabling the use of a reference sensor. To allow the use of an external amplifying circuit the line connectors were chosen since, by default, their programmable amplifiers have unitary gain [57, 58].

The signal samples are transferred to/from the DSP through the Multichannel Buffered Serial Ports (McBSP). The McBSP consists of a data path and a control path that connect to external devices. Separate pins for transmission and reception communicate data to these external devices. Four other pins communicate control information (clocking and frame synchronization).

A 32 bit wide EMIF(External Memory Interface) is used to allow communication between the DSP and an external memory device. The SDRAM, Flash, CPLD and daughtercard expansion connector are all connected to the bus and are memory mapped on the DSP with 4 EMIF addressable spaces.

The CPLD is a programmable logic device used to implement glue logic that ties the board components together, control the daughtercard interface and signals, and allow software control of board features. The user control is achieved with 4 memory-mapped registers mapped into EMIF data space at address 0x90080000. They appear as as 8-bit registers with a simple asynchronous memory interface [59]. In the project the registers were used primarily to access the 4 LEDs, the 4 DIP switches (register 0 - USER_REG), control the daughtercard interface(register 1 - DC_REG) and change the selection bit of McBSP's MUX (register 3 - MISC).

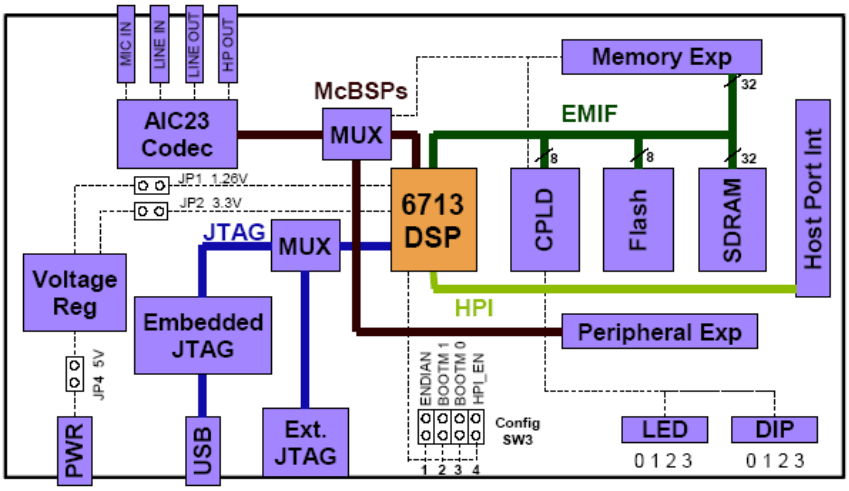


Figure 4.1: C6713 DSK block diagram.

The communication between DSP and CODEC is controlled in interrupt mode, which interrupts the processor at the configured sample rate. In this instant the output sample is sent to the DAC and the input sample is received from the ADC. The libraries which implement the communication and configure the board for use were provided by the daughtercard developers. Data is received from the selected CODEC as two 16-bit words (left/right) packed into one 32-bit word. The developed programs use a

union which allows the data to be accessed as a single entity when transferring to and from the serial port, but still be able to manipulate the left and right channels independently. This union is defined as:

```
union {
unsigned int samples;
short channel[2];
} buffer;
```

4.1.2 DSK_AUDIO 4

This daughtercard, seen in Figure 4.2, is used to increase the number of input and output channels for various models of the TMS320 family of digital signal processor starter kits. It is installed on the C6713's J3 connector, the peripheral interface connector. Included in the connector the following signals are worth mentioning: the CNTL0/1 and STAT0/1 from the CPLD's DC_REG register, used to control the daughtercard; the McBSP signals used to interface and synchronize the DSP with the CODECs.

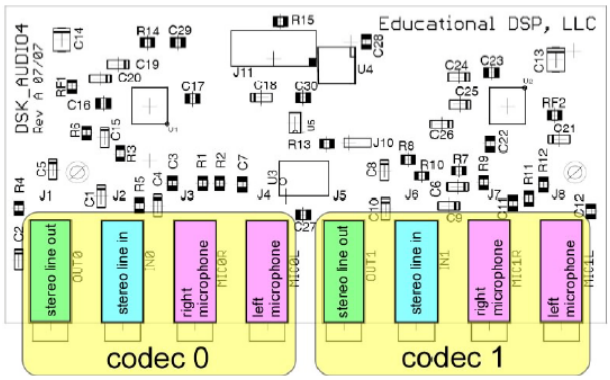


Figure 4.2: DSK_AUDIO4 board layout.

Using two synchronized 16 bit PCM3794 CODEC, operating as slave devices, the card was designed to accept the standard 3.5 mm mini-plugs. There are four audio connections available for each CODEC: one stereo line input, one stereo line output and two mono microphone inputs (biased). The CODECs have a maximum input range of $1V_{rms}$ (± 1.41 V) [58].

Even though the CODEC supports sample rates from 5 to 50 kHz, the libraries supplied by the developers have a selection of pre-configured sample-rates of 16, 24, 32 and 48 kHz. To use the whole range of sampling rates involves some alterations in the daughtercard PCB which could possibly damage the hardware. So, for this reason and the reasons stated in chapter 2 the multirate processing technique was used instead.

The DSK_AUDIO4, seen in Figure 4.2, operates on power-up, by default, at a 48 kHz sample rate using stereo line-level inputs and outputs with 0dB gain. It can be programmed by the DSP application with an application programming interface (API), supplied by the manufacturers, to control various CODEC capabilities such as powering the CODEC, input select, control of the ADC pre-amplifier, DAC amplifier gain and DAC attenuator.

4.1.3 Headset

The headset, containing the sensors (error microphones) and actuators (loudspeakers), is where the noise cancelling occurs. They are the active noise-cancelling headset *ATH-ANC1 QuietPoint*® without the control circuit. By using them, the project is simplified because acoustic precautions, such as space acoustics, error microphone positioning, and loudspeaker and microphone dimensioning, such as necessary sensitivity, durability or frequency response, are disregarded. The disadvantage is that the performance will be limited by these factors.

Table 4.1 shows the speakers characteristics freely available in the product specifications. The speaker rated impedance had to be estimated since it was not available. For that, following procedure described in [60], the minimum impedance of the disconnected speaker was measured with an ohmmeter, obtaining 45 Ω, so a good estimate would be 48 Ω. This value will be used to design the audio amplifier used to drive these speakers.

Table 4.1: Speaker specifications.

Parameter	Value	Units
Sensitivity	105	dB SPL/mW
Driver Diameter	36	mm
Frequency Range	10 - 20,000	Hz
Maximum Input Power	500	mW
Impedance	48	Ω

The microphone should be an electret microphone due to the widespread use of these types of microphones in ANC headsets [12]. A typical capsule has the junction field effect transistor (JFET) in common source configuration so the electret microphone needs to be biased. That involves choosing a proper bias resistor, R_B, and use it to connect the power supply, V_+, to the JFET's drain (positive lead of the capsule) to power the transistor and convert the varying current into a voltage. Also, AC coupling has to be done with a capacitor to block the DC voltage, only letting the AC signal to pass to the next stage. This inclusion not only makes the following stage easier to bias, but it also reduces any noise which might exist in the frequencies below those of interest [61].

The microphone specifications were not accessible, so some measurements were done to estimate some of the necessary values for dimensioning, such as the sensitivity and the operating voltages where the sensitivity drops. During operation, the JFET must be in the saturation region of its characteristic V(I) to properly work as an amplifier. If the voltage drop goes below the saturation voltage $V_{\text{DS}_{\text{sat}}}$ a sensitivity drop is expected. To determine $V_{\text{DS}_{\text{sat}}}$ the microphone was biased with a 2.2 kΩ load resistor, a 1 kHz sinusoidal sound was emitted and the supply voltage was decreased until a sensitivity drop was observed in the oscilloscope. The sensitivity drop was observed at $V_{\text{DS}} = 1$ V. It should be noted that the JFET generally has a maximum voltage rating of V_{MAX} [62, 63].

With this value obtained it is finally possible to dimension the bias resistor R_B. The steady state

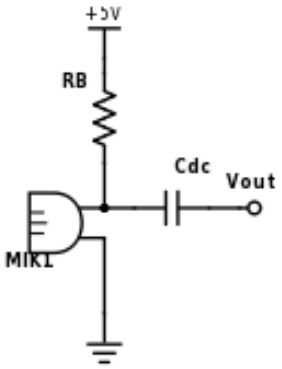


Figure 4.3: Electret microphone biasing circuit.

equation of the circuit depicted in Figure 4.3 when the transistor is in saturation is

$$V_+ = R_B I_{\text{DS}} + V_{\text{DS}}[V], \tag{4.1}$$

where I_{DS} V_{DS} are the current and voltage drop through the transistor, respectively. The supply voltage chosen was 5 V, with the reason being stated in subsection 4.1.6. The bias point was set to the middle of the voltage range, from $V_{\text{DS}_{\text{sat}}}$ to the supply voltage, in this case $V_{\text{DS}} = 3$ V, to maximize the dynamic range. The current used for dimensioning was the usual maximum current of these devices, $I_{\text{MAX}} = 0.5$ mA [62, 63]. This way, if there are variations on the JFET saturation current, as a result of external influences (temperature, fabrication defect...), the voltage drop on the JFET does not fall behind the saturation voltage, reducing sensitivity or turning off the FET. Resolving Equation 4.1 for R_B it returned the value 4 kΩ, being the 5.1 kΩ the resistor value available in the laboratory stock.

If the following amplifying stage has high input impedance, the high-pass filter formed by the bias resistor and the capacitor will have a very low cut-off frequency as seen in the following equation

$$f_c = \frac{1}{2\pi R C_{\text{DC}}}[Hz], \tag{4.2}$$

where R is the resistance seen by the capacitor C_{DC}, $R = R_B + Z_{\text{in}_{\text{pre}}}$, where $Z_{\text{in}_{\text{pre}}}$ is the input impedance of the amplifier. The capacitor used was readily available in the laboratory stock, being $C_{\text{DC}} = 1\mu$ F. As will be stated, the input impedance of the pre-amplifier is $Z_{\text{in}_{\text{pre}}} = 50$ kΩ, which will give a cut-off frequency of $f_c = 3$ Hz.

The following table summarizes the microphone bias circuit design values. In operating conditions the measured current was 0.2 mA, the voltage drop across the electret microphone was 3.80 V and no sensitivity drop was observed.

With the microphone powered, its sensitivity was estimated. To achieve that an additional microphone (later used as the reference microphone), with its specifications known, was used [62]. It was positioned at a distance of 50 cm and a height of 5 cm from the base of the noise source loudspeaker, aligned with

Table 4.2: Microphone in Steady State

	Value	Units
V_+	5	V
V_{DS}	2.5	V
I_{MAX}	0.5	mA
R_B	5.1	kΩ
C_{DC}	1	μ F

the centre of the loudspeaker.

Then, a sinusoidal sound with 1 kHz was emitted and its loudness increased, so the reference microphone would output the amplitude correspondent to a sound with 1 Pa (94dB SPL), measured by an oscilloscope. According to its datasheet, the sensitivity of -42 ± 2 dBV/Pa corresponds to 7.94 ± 2 mV/Pa so, a value around this was expected. Then the ear cup was positioned in the same way with the same sound and the estimated sensitivity obtained for the headset microphones was $-37,59 \pm 2$ dBV/Pa (13.2 ± 2 mV/Pa). This experimental procedure, specifications and units are defined by the IEC 60268-4 norm [64, 65].

4.1.4 Signal Conditioning Circuit

Considering a sinusoidal sound with frequency of 1 kHz with the maximum noise intensity of 100 dB SPL (or 2 Pa) the signal at the output of the microphone has an amplitude of about 26,4 mV. Before delivering it to the line level input it is desirable to bring the full-scale range of the signal to, at least, half the full-scale range of the ADC. That is because if a signal does not occupy that amount of the full input voltage range of the ADC, the signals acquired suffer a severe performance hit in terms of effective ADC bit resolution.

The CODEC used has programmable pre-amplifiers included in the package however, an analogue pre-amplifier circuit was designed to meet the objectives of the project. This voltage amplifier, represented in Figure 4.4, was developed with the non-inverting amplifier as a starting point. Then, biasing was added to the non-inverting input and the inverting input was decoupled as described in the following paragraphs.

The non-inverting topology was chosen instead of an inverting topology as this last one has low input impedance, to meet the requirements stated in the previous subsection. To find out how much voltage gain the pre-amplifier needs, the ratio between the ADC full-scale range and the input signal swing (difference between maximum and minimum value) was computed, resulting in $A_{V_{\text{pre}}} = 55$. Then, a 200 kΩ potentiometer was chosen for the feedback resistor, to have low currents on the feedback loop. With this information, the voltage gain expression of a non-inverting amplifier was solved for R_3 giving

$$A_{V_{\text{pre}}} = 1 + \frac{R_f}{R_3} \tag{4.3}$$

$R_3 =$ 3.6 kΩ. A potentiometer was used to enable adjustments on the gain in order to stabilize the final

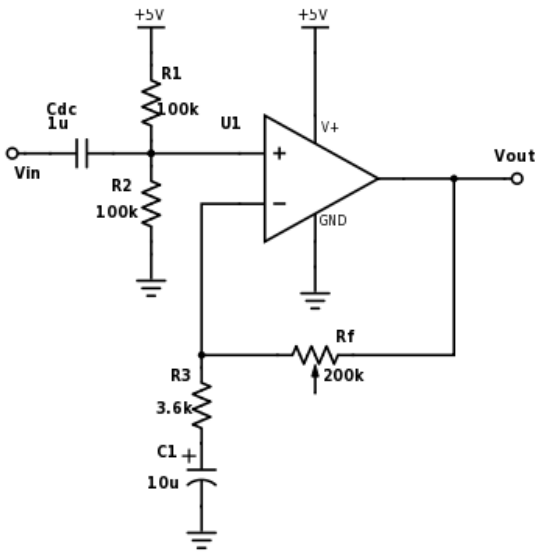


Figure 4.4: Pre-amplifier Circuit.

circuit in case of unwanted oscillation. However it is expected to be set near its maximum position.

Then, the signal should be brought up to a bias point which, in AC amplifiers, is the centre point where the output will oscillate around. Considering a single-supply at 5V, which positions the linear region of the amplifier between ground and the supply voltage, the bias used was mid-supply point, 2.5 V because it keeps the OPAMP from saturating (amplifying a signal beyond the supply voltage limitations) and guarantees the signal swing to be as large as possible. Thus the output signal will oscillate between [1.09, 3.91] V.

To achieve this, a voltage divider was made with two equal valued resistors and connected to the supply. The only precaution needed was to ensure a corner frequency of the hi-pass filter formed by the DC-block capacitor and the resistors parallel to be much lower than the lowest cut-off in the associated circuit, specifically 30 Hz, and at the same time ensure low power consumption. So, with R_1,$R_2 = 100k\Omega$, chosen to achieve low currents and maintain high input impedance, the resulting corner frequency would be an acceptable 3 Hz [61].

However this topology has a problem since the DC voltage is amplified as well thus saturating the OPAMP. To avoid this, a capacitor was included between R_3 and ground, thus changing the circuit to a buffer amplifier in steady-state, meaning the gain for the DC component is unitary. The capacitor value was chosen to maintain the RC relation of the biasing section. Solving equation 4.4 one obtains $C_1 =$ 10 μF.

$$f_c = \frac{1}{2\pi R_1 || R_2 C_{\text{DC}}} = \frac{1}{2\pi R_3 C_1}[Hz] \tag{4.4}$$

The use of low currents is stressed, to maintain the power consumption of the circuit low. What remained to do was to AC couple the circuit with the next stage, the ADC, however, as aforementioned, this is already done at the board terminals, by the manufacturers.

The device used was the MCP6286T-E/OT, from Microchip Technology Inc., for its small size pack-

age, SOT-23-5, and power supply range including the 5 V [66].

4.1.5 Reference Microphone

A reference microphone and its pre-amplifier circuit was developed to capture the reference signal in order to allow the testing of algorithms that use the feedforward configuration. The same procedure as the previous two subsections was followed, however to obtain the half of the full-scale range was enough. The electret microphone ABM-713-RC, from Pro-Signal, and the LF412CN OPAMP, from National Semiconductor, were used. The circuit was assembled on a breadboard since electrical noise was not expected to be an issue. This was because the level of the pre-amplifier output signal would be much higher than the noise, not influencing the ANC system results. However, this option could reduce the coherence between the reference signal and the primary noise captured by the error microphone due to the expected electrical noise.

4.1.6 Audio Amplification Circuit

As stated in 4.1.1 the DSP interfaces with the loudspeakers through the line-out connector. Line level is the specified strength of an audio signal used to transmit analog sound between audio components and is intended to drive a load impedance of 10 kΩ such as CD and DVD players, television sets, audio amplifiers, and mixing consoles. This means that line-out is expected to deliver minimal current with only a few volts, however to drive a low impedance load, like the 48 Ω speakers, an audio amplifier must be used to inject the current the load requires, i.e. lower the output impedance of the system.

An audio power amplifier IC was adopted to reduce the number of required passive components required and the complexity of the project. This is the TS4871ID, from STMicroelectronics, with a SO8 package and 5.5 V maximum supply. It is a monolithic power amplifier with a BTL output type. BTL (Bridge Tied Load) means that each end of the load is connected to two single ended output amplifiers. Thus, one of the outputs gives a negative voltage $V_{\text{out1}} = V_{\text{out}} = -V_{\text{in}}\frac{R_{\text{feed}}}{R_{\text{in}}}$ and the other a positive voltage $V_{\text{out2}} = -V_{\text{out}}$. When connected to their respective speaker terminals the amplifier's output is $V_{\text{out1}} - V_{\text{out2}} = 2V_{\text{out}}$ [V]. Thus, the amplifier output power is [67]:

$$P_{\text{out}} = \frac{(2V_{\text{out}_{\text{rms}}})^2}{R_L}[W], \tag{4.5}$$

meaning that for the same power supply voltage, the output power in BTL configuration is four times higher than the output power in single ended configuration.

Resorting to the typical application circuit, depicted in Figure 4.5, and to the datasheet, the essential components for this application were narrowed down. For example, the resistor for the stand-by switch was ignored since there was no need for a stand-by switch. Another component disregarded was the power supply bypass capacitor C_s since its influence on THD+N was on frequencies above 7 kHz (outside the band of interest), it affected indirectly on power supply disturbances and the recommended value was too high.

The power amplifier dictated the power supply voltage. Considering the high impedance of the

speakers, in relation with the recommended impedances in the datasheet [67], a value near the maximum rated supply voltage, 5 V, was used in order to maximize the power delivered by the amplifier to the load to avoid a lack of energy to cancel noise.

To determine the maximum gain of the amplifier, first V_{out} was calculated with Equation 4.5, knowing the maximum power the loudspeakers can handle, 500 mW, their rated impedance and the input signal maximum voltage. It was obtained $V_{\text{out}} = 2.45V_{\text{rms}}$. Doing the ratio between this and the input voltage gives

$$Gv = \frac{-V_{\text{out}}}{V_{\text{in}}} = 2\frac{R_{\text{feed}}}{R_{\text{in}}} = 2.45. \tag{4.6}$$

Note that the maximum level of the CODEC is $1V_{\text{rms}}$.

To size the resistors a trade-off must be accounted for, considering the high-pass filter formed by C_{in} and R_{in} and the low-pass filter by C_{feed} and R_{feed}, to obtain reasonable values for the capacitors and at the same time cut frequencies outside of the audible range. To calculate the cut-off frequencies of the resulting filters, the following expression

$$f_{c_{\text{LP}}} = \frac{1}{2\pi R_{\text{feed}} C_{\text{feed}}}[Hz], \tag{4.7}$$

was used for the low-pass filter with $f_{c_{\text{LP}}} = 20000$ Hz, and

$$f_{c_{\text{HP}}} = \frac{1}{2\pi R_{\text{in}} C_{\text{in}}}[Hz], \tag{4.8}$$

was used for the high-pass filter with $f_{c_{\text{HP}}} = 20$ Hz. The R_{in} and C_{in} values were found, by fixing $R_{\text{in}} = 100k\Omega$ to get a high input impedance, thus obtaining $C_{\text{in}} = 82$ nF from Equation 4.8 respectively. For R_{feed} a 200 kΩ potentiometer was used to experimentally fine-tune the gain according to the system's demands, however C_{feed} was calculated with the maximum gain in mind, meaning that with 120 kΩ a 49 pF was the closest common capacitor value.

This amplifier needs two bypass capacitors. One of them was not used, as already declared, however C_b has influence on THD+N in lower frequency. The recommended value of 1 μF was used and the "Pop and click performance" was evaluated: if the resulting value of expression 4.9 is much lower than C_b capacitance, pop and click effects will be reduced. The value obtained was 0.361 μF and since this is less than half of C_b, the previous condition is met [67].

$$C_b \gg C_{\text{in}}\frac{R_{\text{in}} + R_{\text{feed}}}{50k\Omega}, \tag{4.9}$$

The following table has all the component's values and the corner frequencies for the filter where it applies.

The previous calculations were made to obtain the maximum values of the components. To set the power amplifier gain Gv to its operating value, an experimental approach was adopted. Then, to verify if the value was enough to cancel the noise levels considered, some calculations were made the speaker sensitivity and rated impedance.

Table 4.3: Audio Power amplifier design.

Component	Value	f_c
R_{in}	100 kΩ	19.4 Hz
C_{in}	82 nF	
R_{feed}	120 kΩ	27.1 kHz
C_{feed}	49 pF	
C_b	1 μF	

The process involved connecting the audio amplifier and pre-amplifier to the DSK. Then, a sinusoidal signal with a quarter of the full representation range and 65 Hz frequency, the lowest frequency of band of interest, was internally generated by the DSP and output to the audio amplifier. The potentiometer R_{feed} was increased until the amplitude of signal captured by the error microphone matched that of the sinusoidal signal with amplitude of 8192. This way, the noise generated has enough energy to cancel the noise detected. A value of $R_{\text{feed}} = 6$ kΩ was obtained for both channels corresponding to a gain of $Gv = 0.12$. The aforementioned conditions about the capacitors remain valid, as the value of R_{feed} was lowered the low-pass filter corner frequency increased and the C_b is even bigger than the value returned by Equation 4.9.

As explained on subsection 4.1.4 the system was calibrated for a maximum expected noise of 2 Pa(100 dB SPL) intensity, meaning the CODECs will output its maximum level, $1V_{\text{rms}}$, when the error microphone senses that noise level. As presented above, the speakers sensitivity is given in dB SPL/mW [68] so the following expression translates that:

$$soundout = sensitivity + 10log(\frac{P_{\text{Load}}}{0.001})[dB_{SPL}]. \tag{4.10}$$

So, to cancel the maximum expected noise, the speakers should output a signal with 100 dB SPL and, resolving Equation 4.10, draw $P_{\text{Load}} = 0.316$ mW. With the gain obtained and knowing the expression for the power used at the load [67]

$$P_{\text{Load}} = \frac{(GvV_{\text{in}})^2}{2R_L}[W], \tag{4.11}$$

the maximum signal output by the DAC will make the load draw 0.311 mW. This value is reasonable, so it confirms the proper calibration of the system.

4.1.7 PCB Development

The circuit was realized in a printer circuit board (PCB) to reduce the audible electrical noise present on the breadboard prototype and to reduce the size of the circuit.

Schematic design

To prepare the board to accept any power supply (ideally a 9 V battery) a voltage regulator was added to the circuit as a fixed-output regulator. Switching converters could be an alternative since they are highly efficient and can also step up the voltage, thus requiring a smaller voltage (the original

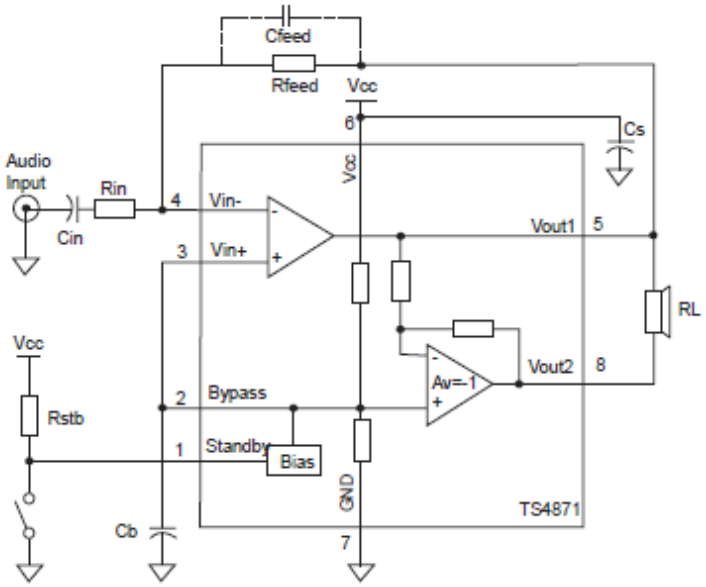


Figure 4.5: Typical application circuit of audio power amplifier [67].

controller of the headset used one). However they are expensive, take a lot of design effort and are noisy [69]. Especially for this last reason, the linear regulator is a superior choice for audio applications than switching converters. The device used was the UA7805CKCT, from TI, with TO-220 package. Following the datasheet [70] typical application an input supply capacitor with 0.33 μF, recommended for filtering noise on the input, and an output supply decoupling capacitor with 0.1 μF to stabilize the output were used. To output the desired 5 V at least 7 V should be in the input.

The complete PCB circuit, depicted in Figure B.2 in Appendix B, included all the mentioned circuits and was made in the industry standard software suite Altium Designer. Looking at the schematic some alterations to the circuits described previously may be noticed. They are the inclusion of 0.1 μF bypass capacitors to the pre-amplifiers, C6 and C9, with the value chosen to bypass mid to high frequency noise[71]; a bypass capacitor space, C3; a shunt (resistor with 0 Ω) between the power rails of the pre and audio amplifiers. These alterations were made as a safeguard against a behaviour observed on the breadboard prototype: when connecting the audio power amplifier and the pre-amplifier circuits to the same power supply the resulting circuit oscillated.

The pre-amplifiers' bypass capacitors, located as near as possible to the ICs power pins, suppress coupling via the power supply connections and reduce any noise added from the power supply to the PCB. The decoupling capacitor space, C3, was included to protect the circuit from signals originating in the power supply and jumper wires and manipulate the bandwidth of the circuit, if necessary, and to prevent cross talk from the power circuit to the pre-amplifier. If, after assembly of the PCB, the decoupling capacitors were not enough to prevent the oscillating circuit, one could cut the shunt, separating the circuits completely, and add an additional power supply to the circuit.

It should be noted that the components used were SMD resistors and capacitors with the exception being the 10 μF capacitors. These were regular aluminium polarized electrolytic with through-hole leads since they were readily available in the laboratory stock and their SMD counterparts were not included

in the bill of materials.

In Appendix B the circuit bill of materials (with the components dimensions) is presented.

Layout design

After completing the schematic for the PCB design its connections and components footprints were imported to the layout tool of Altium Designer. The PCB layout for top and bottom layers is shown on Figure B.1 in Appendix B. A photograph of the top layer of the assembled circuit is presented in Figure 4.6.

The right and left channel circuit blocks, each composed of a pre-amplifier and audio amplifier circuits, were placed symmetrically with their components in a configuration to have the shortest signal traces. Also they have the most space possible between them in order to occupy most of the available area offered by the PCB manufacturer, to ease the soldering process.

The power paths were given larger width (double of the signal traces) in order to reduce the trace's resistance, reducing the heat caused by dissipation. This is because all the current is supplied by those wires and failing to do so could generate excessive heat and burn the wires.

The PCB has two layers, the top and bottom layers. Copper pours were done on both layers and connected to ground, after placement of the devices footprints and routing. These "ground planes" are done especially in analogue circuits in order to have a consistent potential for ground throughout the PCB. If traces were used as the signal's return paths, their resistance would create voltage drops that would generate different grounds in the PCB. Normally a whole layer should be reserved for this purpose in order to avoid breaks on the ground paths. The copper pour on the top layer was done as a result of good practice, since using a lot of area reduces the usage of etchant, reducing costs in manufacture (even though only for large production volumes). Another reason is better noise shielding and signal contention, even though this reason is more important in RF circuits. These copper planes also improve heat dissipation on the PCB.

Another good practice to reduce noise, help maintain a low impedance and short return loops is the usage of a technique called via stitching [72]. Also more critical in RF circuits, it still is useful in audio circuits with a lower via density.

The main consideration during PCB assembly was to start with the smaller (and more abundant) components such as the resistors and capacitors and then soldering the ICs and to be wary of the proper polarities of the electrolytic capacitors.

4.2 Software

Software development on the DSK is done in TI's Code Composer Studio (CCS) software suite, version 6.1.0. This integrated development environment (IDE), based on the Eclipse software framework, includes all required tools such as the assembler, compiler, linker and debugger. It communicates with the boards through the on-board JTAG emulator. To simplify the explanations, a single-channel ANC system will be considered but the headset is a two channel system, therefore all of the processing operations

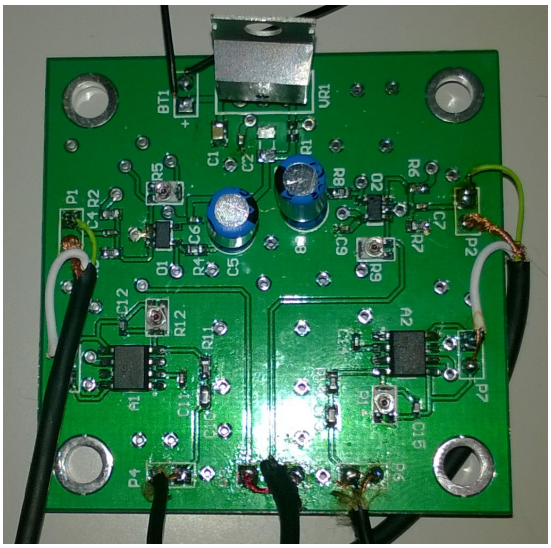


Figure 4.6: Assembled analog circuit.

were made on samples from each channel. It should be noted that, CODEC0 of DSK_AUDIO4 is used to input the error signals and output the control signals. CODEC 1 is used to input the reference signal for feedforward algorithms.

4.2.1 Software overview

The programs developed start with an initialization routine before the ANC algorithms. This routine includes commands to setup the board to use the daughtercard such as reseting the daughtercard, redirecting the McBSPs to the J3 connector (peripheral interface connector), configuring the interrupts and sample rate among others. Then, the arrays for the filters coefficients, which are $\mathbf{w}(n)$, and the filters buffers, which are $\mathbf{x}(n)$, are initialized and there is a 4000 cycle wait for the board to start up. During the waiting period, samples valued zero are sent to the CODEC to avoid acoustic feedback, since the CODEC output is connected to its input by default.

As stated, every sampling period T_s the program is interrupted and the input/output routine is called. In it, the samples output by the ADCs are received from the McBSP. More specifically, there is a data structure which contains variables associated to all the McBSP signals, and the contents of the variables used for data transfer are copied to the input serial port array. Afterwards, the multi-rate processing routine is called which will process the sample from the ADC and a sample from the ANC algorithm, if this last one is available. The output serial port array contents are transferred to the McBSP, to be sent to the DAC. Each time the interrupt routine is called, a counter is incremented and an if-clause is used to verify the counter value in order to change the state of a flag, also related to multi-rate processing.

After the waiting period, the program enters an infinite loop. The control flow stops at an if-clause, which checks the state of the flag changed in the interrupt routine. When this flag is turned on, the ANC algorithm is started and the flag is turned off again. Then the algorithm the program returns to the if-clause.

There are two function constantly used in this program, the FIR filtering and the estimation of the

input signal power of the adaptive filter, used in the NLMS as explained in subsection 2.2.2. FIR filtering has two processes involved, the actualization of the buffer and then the convolution. When the buffer actualizes, all the existing samples are shifted to the right to make space for an arriving sample. Then the filtering is executed with a convolution, represented in Equation 2.1, which is the summation of the multiplication of the filter ith coefficient with the ith buffer position. To implement the estimation of the input signal power, computed with the square of the euclidean norm, the summation of the multiplication of the ith buffer position with the ith buffer position is used.

4.2.2 Multi-rate signal processing

To exploit the properties of high and low sampling rates the multi-rate signal processing technique was employed. As mentioned, ANC is most useful at low frequencies, generally bellow 500 Hz, thus that was considered as the band of interest for this project. One could use a sampling rate twice the limit of the band (Nyquist rate) however, the resulting decimation and interpolation filters would have to be very sharp and very high order would be needed. To accommodate for lower order filters, a sampling rate F_s of 2 kHz was chosen for the ANC algorithm operation. The sampling rate of the CODEC, was chosen for the same reason. If a high sampling rate was used then a large sampling frequency variation would be required resulting in very high order decimation and interpolation filters. Therefore, the lowest sampling rate pre-configured in the daughtercard libraries, $F_s = 16$ kHz, was chosen. Using these sampling rates results in a decimation factor of $M = 8$ and an interpolation factor of $L = 8$.

The decimation was implemented by filtering each new acquired sample, contained in the input serial port array, with the decimation filter. The filter output is then copied to an array which serves as an input to the ANC algorithm, when it is called every M samples. The decimation is also applied to the reference signal, in feedforward algorithms, and its output is also sent to the ANC algorithm.

The interpolation was implemented with a counter and modulus operator which would verify if the current iteration is a multiple of L. If the current iteration is a multiple of L, the output of the ANC algorithm would be filtered by the interpolation filter, otherwise a zero sample filtered. Then, the filter output would be sent to the output serial port array.

Since the conversion factors are equal, the same filter coefficients can be used for both anti-aliasing and anti-image filters. They were low pass filters with a passband cutoff frequency of 500 Hz and a stopband corner frequency of 1 kHz. As stated, the resulting filter order would be lower than with closer cuttoff and stopband frequencies, because the transition band would be bigger. The ”fdatool” from MATLAB’s Signal Processing toolbox was used to compute the filter coefficients with a kaiser window and linear magnitude specifications with Dpass = 0.0001 and Dstop = 0.0001. When choosing the ripple there was a trade-off between the preservation of information of the signals and the increase of the filter length and delay. The adaptive filter may compensate the ripple of the passband, but that is not a given. The stopband ripple generates background noise. Therefore the ripples were chosen very small to decrease these effects and the transition band was the biggest possible to decrease the filter order, compensating the ripples chosen. The resulting filter is represented in Figure 4.7, it is a 161st order FIR lowpass and it operates at 16 kHz.

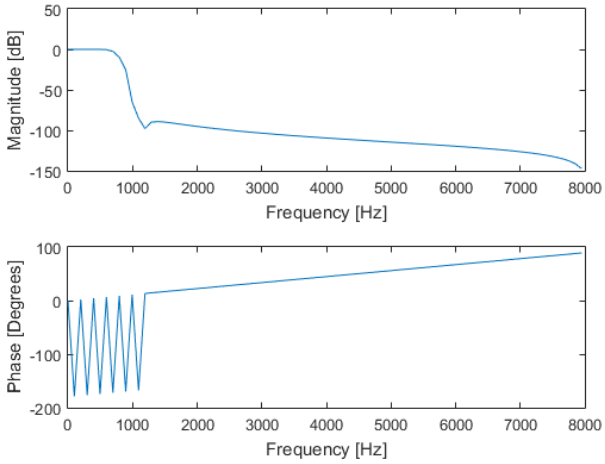


Figure 4.7: Frequency response of Anti-aliasing/Anti-image 161st order lowpass FIR filter.

The ANC algorithm can only start a new iteration every $M = L = 8$ interruptions. At that time it receives a sample from the decimator and it must have a new control signal sample ready for the interpolator. As stated, a flag was used to signal these instants and was used in an if-clause to control the call to the ANC algorithm. A counter counts the number of interruptions and a if-clause with a modulus operator verifies if the counter has reached a multiple of the conversion factor.

The ANC algorithm runs while the interrupt routine is not called. Since the frequency conversion functions are called during the interrupts this means the ANC algorithm runs in parallel to them and has $M = L = 8$ pre-decimation sampling periods to make its computations. This is possible to do because at the end of an interrupt routine the program returns to the point it left off. The acquisition and processing are naturally synchronized as long the processing time of the noise cancelling algorithm is less than eight times the pre-decimation sampling period, $t_p < 8T_s$, or one post-decimation sampling period. However a better estimate for this period includes the time the conversion functions takes, t_d for decimation time and t_i for interpolation time, which takes time away from the available time for the processing, $t_p < 8(T_s - t_d - t_i)$.

4.2.3 Offline secondary path estimation

As mentioned in the previous chapter, an ANC system has the secondary-path following the control filter. To compensate this the FxLMS algorithm is used, but it needs an estimate of the secondary path to do so. In this project both off-line and on-line methods of secondary-path modelling were used. On-line modelling is an integral part of the ANC algorithm so, its implementation depends on the algorithm design. However, off-line modelling is always executed at the beginning of the FxLMS algorithms, thus its implementation on this project will be described in this subsection.

This process is a system identification problem. An iteration starts by receiving the output of the decimator $d(n)$, which is the training signal passed through the secondary-path. Referring to Figure 2.11, the training signal sample $x(n)$ is generated internally by the DSP, output to the speaker and then captured by the error microphone. An estimate of $d(n)$ is computed with the training signal filtered by the

current secondary-path impulse response estimate $\hat{s}(n)$, $y(n)$. Then, an error signal $e(n)$ is computed with the difference between $d(n)$ and $y(n)$. The error signal and training signal are input to the LMS algorithm to update the weights of $\hat{s}(n)$. After an amount of time the error signal should converge to the background noise and the secondary path estimate to the secondary path and then it can be used in the FxLMS algorithm or its variants.

A white noise was used as the training signal since it has constant spectral density at all frequencies. For that a linear congruential sequence generator was implemented. It is an algorithm that yields a sequence of pseudo-randomized integers by iteratively solving the recursive expression

$$I_{n+1} = (JI_n + 1) \mod M, \tag{4.12}$$

where the "modulus" M= 2^{20}= 1048576, the "multiplier" J = 2045 and the "start value" I_0 = 12357 were the values used as suggested in [3]. I is the sequence of pseudo-random values returned.

This algorithm returns integers in the interval [0,2^{20}] which are well beyond the CODEC numerical range. To obtain values in the interval [-0.5,0.5] the output random sample was normalized with M as the normalizing factor and centred around zero by subtracting it with 0.5. Then a digital gain of G_d = 10000 is used to bring the random noise to audible levels.

The NLMS algorithm was used for the weight update. The off-line modelling step size μ_{offmod} was normalized by the training signal power,

$$\mu_{\text{offmod}} = \frac{\mu_s}{P_x}, \tag{4.13}$$

with μ_s = 0.01.

4.2.4 FxLMS algorithm

After the off-line modelling is completed, the system is operated in the active noise cancellation mode. As stated, in feedforward algorithms the reference noise is obtained from the reference microphone. In feedback algorithms, the reference signal is obtained from an estimate of the primary noise, which is computed with the difference between the error signal, $e(n)$, and the control signal of the previous iteration, $y(n-1)$, passed through the secondary path estimate, $\hat{s}(n)$, as in

$$\begin{aligned} x(n) &= \hat{d}(n) \\ &= e(n) - \hat{y}(n-1) \\ &= e(n) - \hat{s}(n) * y(n-1). \end{aligned} \tag{4.14}$$

Then a control signal sample is generated by filtering the reference signal through the current control filter, $w(n)$, inverting it and sending to the interpolator, $y(n) = -x(n) * w(n)$. The reference signal is then filtered by the secondary path estimate to generate the filtered-x signal, $x'(n)$, and its power is computed. The $x'(n)$ and error signal are input in the NLMS algorithm to update the control filter coefficients. The adaptation step size is normalized with $x'(n)$ power computed in the previous step. The program then loops and waits for the next iteration start.

I should be noted that the filter order, step size and leakage factor were determined experimentally to maximize attenuation. The results are presented in chapter 5.

4.2.5 Band Limited MFxLMS algorithm

Off-line modelling is conducted first to estimate the secondary-path transfer function. The noise canceling mode begins after the training.

The algorithm is similar to the FxLMS except for the adaptive algorithm inputs. In the MFxLMS algorithm the inputs for the NLMS algorithm are the filtered-x signal $x'(n)$ and the modified error signal $e_m(n)$. This last one is computed with the difference between the desired signal estimate and the output of the control filter with the filtered-x as input, $e_m(n) = \hat{d}(n) - w(n) * x'(n)$. Since it is an algorithm applied to a feedback system, the reference signal is obtained as in the feedback FxLMS. Thus the desired signal estimate $\hat{d}(n)$ and the reference signal $x(n)$ are the same.

To accomplish the band limiting, the reference signal should be filtered by a bandpass filter before computing the $e_m(n)$ and before the secondary path estimate filter to obtain the filtered-x signal. This way $e_m(n) = f(n) * \hat{d}(n) - f(n) * x(n) * \hat{s}(n)$.

As will be seen in chapter 5, the phase error is above 90° for frequencies under 80 Hz. The bandlimiting filter, $f(n)$, should be flat in the pass band and have a constant group delay, so it was implemented using a linear phase FIR filter. It was designed with MATLAB "fdatool" and window method using a kaiser window, with the post decimation rate in mind. A filter with specifications to only remove the problematic frequencies, with a small transition band and low ripple returned a very large length. So, to obtain a larger transition band the band of interest was shortened and the stopband ripple was increased to obtain a lower filter order. The obtained filter specifications were the following:

Table 4.4: Bandlimiting filter design specifications.

Parameter	Value	Units
F_s	2000	Hz
Fstop1	25	Hz
Fpass1	100	Hz
Fpass2	500	Hz
Fstop2	900	Hz
Dstop1	0.01	Linear
Dpass	0.1	Linear
Dstop2	0.01	Linear

It returned a filter with length, N=61. Its frequency response can be observed in figure 4.8.

4.2.6 MMFxLMS algorithm

This algorithm is a derivation of the MFxLMS using the OMA algorithm to model the primary and secondary path. The implementation followed closely the algorithm presented in [24].

With a new reference signal sample the desired signal estimate is computed with the estimate of the

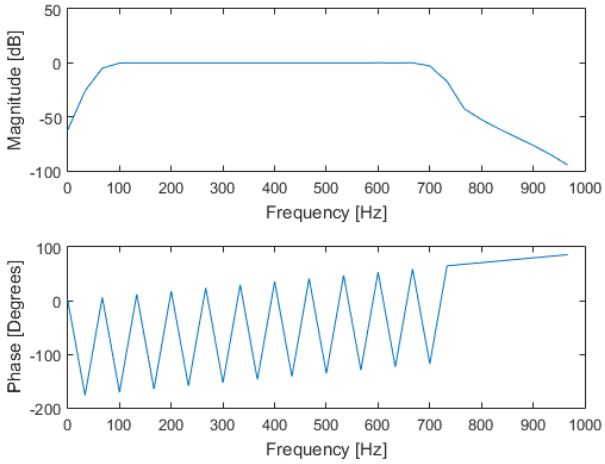


Figure 4.8: Frequency response of bandlimiting 60th order bandpass FIR filter.

primary path of the previous iteration, $\hat{d}(n) = \hat{p}(n) * x(n)$. The new anti-noise sample is computed and sent to the speakers, $y(n) = -x(n) * w(n)$.

Then an estimate of the error signal is computed with the estimates of the primary and secondary path, $\hat{e}_1(n) = \hat{p}(n) * x(n) + y(n) * \hat{s}(n)$. The OMA error signal is computed with the difference between the error signal and its estimate, $\alpha(n) = e(n) - \hat{e_1}$. Then the primary path and secondary path estimates are updated with the NLMS, with the normalization factor being given by the sum of the control signal $y(n)$ and the reference signal $x(n)$ powers. To update the first $x(n)$ and $\alpha(n)$ are used as inputs, to update the second $y(n)$ and $\alpha(n)$. The step-size used for this weight update is the same as the off-line method, $\mu_s = 0.01$.

The inputs for the NLMS of the control filter are the same as in the MFxLMS, the filtered-x signal and the modified error signal, $e_m(n) = \hat{d}(n) - w(n) * x'(n)$. The normalization factor is still the the filtered-x signal power. This means that the modified error signal is another estimate of the actual error signal.

Chapter 5

Results

This chapter presents the results of various real-time tests to understand the capabilities of the system. All the tests were realized in a room at INESC-ID with the surrounding environment resembling real-life settings, with ambient noise and sound reflections from various objects in the room. All the experiments had a user wearing the headset and the data was acquired by plotting the system's signals in the graphing tool of CCS, exporting that data and process it in MATLAB.

In sections 5.1 and 5.2 the secondary paths of the system are characterized. Then, each subsequent section will focus on one of the used algorithms. Each algorithm went through a process of changing various parameters, with the resulting performance assessed. The results of these experiments are always presented in the first subsection of each section, served to optimize the system performance by obtaining the best combination of parameters to be used on the following analysis. During it, different primary noises were used; first a single sinusoidal tone with rising frequencies, then a narrowband noise of an industrial compressor and lastly a broadband noise. The performance results of those tests are shown in the second subsection of each section. Then those results are compared to some literature results and discussed in section 5.7.

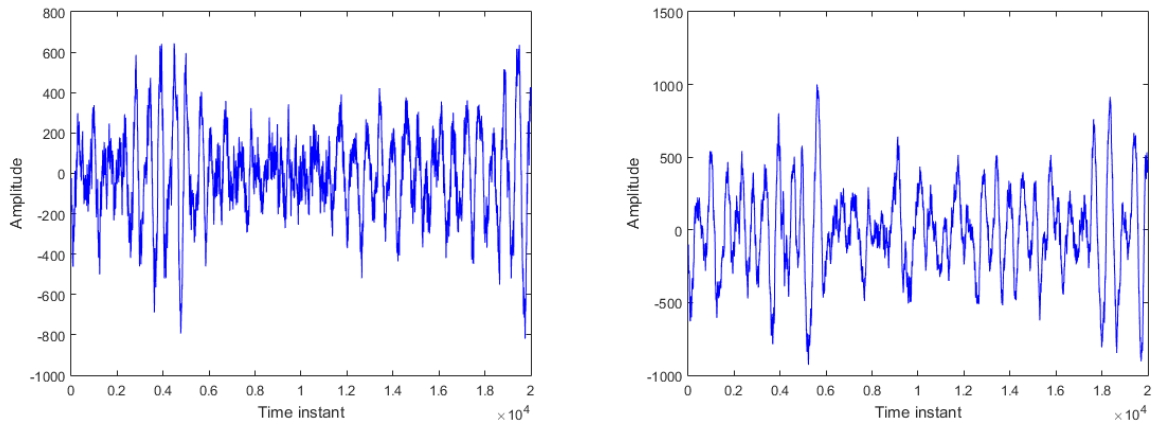
5.1 Audio system measurements

The audio system consists of the stereo headset, DSK board and analogue amplifier circuit, making the secondary paths of both channels. To know to what extent the hardware affects the system performance, some of the system's audio characteristics were measured. Unless otherwise noted, all data for this examination was obtained with the following parameter values:

- Secondary path off-line modelling NLMS step size $\mu_s = 0.01$;
- Leakage factor $\lambda = 1$;
- Sample Rate $F_s = 16$ kHz;
- $\hat{S}(z)$ adaptive filter length $L_s = 128$;
- Duration of the off-line modelling $Ns = 10000$;
- Power amplifier gain $Gv = 0.4$;

•Error microphone pre-amplifier gain $A_{V_{\text{pre}}} = 23$.

The ambient noise present with these set of parameters is seen in Figure 5.1. As stated, the CODECs have 16 bits so the numerical range was [-32768,32767]. For this set of parameters, a sinusoidal signal with the maximum noise level considered (2 Pa) has an amplitude of 9970 RMS and the ambient noise approximately 300, therefore the ambient noise is 3% of the maximum noise.



(a) Left Channel (b) Right Channel

Figure 5.1: Ambient noise at the time of audio system measurements.

A block diagram depicting the signals involved and their relation to the secondary path is portrayed in Figure 5.2. It should be noted that they follow the nomenclature used for the off-line secondary path modelling algorithm.

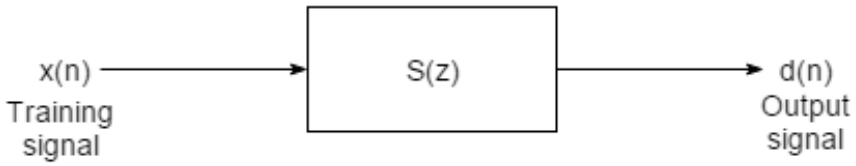


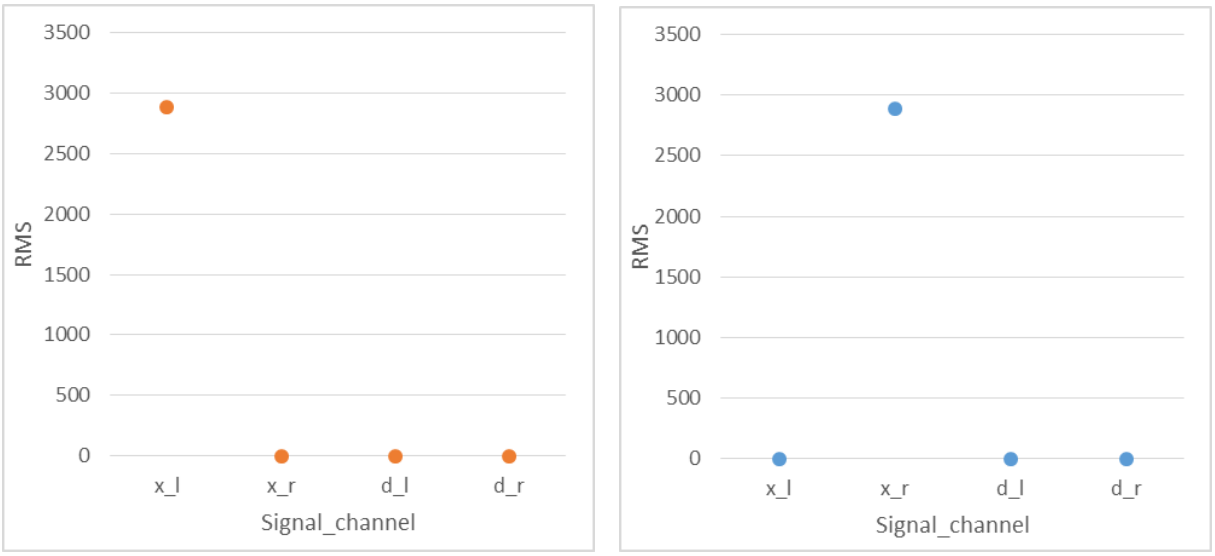
Figure 5.2: Signals used for audio measurements.

5.1.1 Crosstalk

Crosstalk is the introduction of electrical noise (from another signal channel) caused by ground currents, stray inductance or capacitance between components or lines. Crosstalk reduces, sometimes noticeably, separation between channels in a stereo system. To guarantee that the DSK have no internal electrical paths that need to be compensated, a simple measurement of crosstalk was made. This way, it is ensured that each single-channel of the active headset system are separated.

The procedure consisted of plugging off the pre-amplifier output from the line-input of the CODEC, thus turning off the microphone. Then a internally generated white noise signal would be injected on the DAC of one of the channels, and output by the loudspeaker, and the signal acquired by the ADC would be observed. If the right output was excited and the left input had a signal, then the DSK had considerable crosstalk.

As seen in Figure 5.3 the signals captured by the ADC, d_l and d_r, have no visible signal. The excitation signal, x_r and x_l , has an amplitude of nearly 3000 RMS and the amplitudes of the received signals, d_r and d_l, 1 RMS. Therefore there there is no electrical crosstalk and it is confirmed that the headset is made of two isolated single-channel systems.



(a) Left Channel Excitation

(b) Right Channel Excitation

Figure 5.3: Electrical crosstalk in analogue circuit.

5.1.2 Non-linear distortion

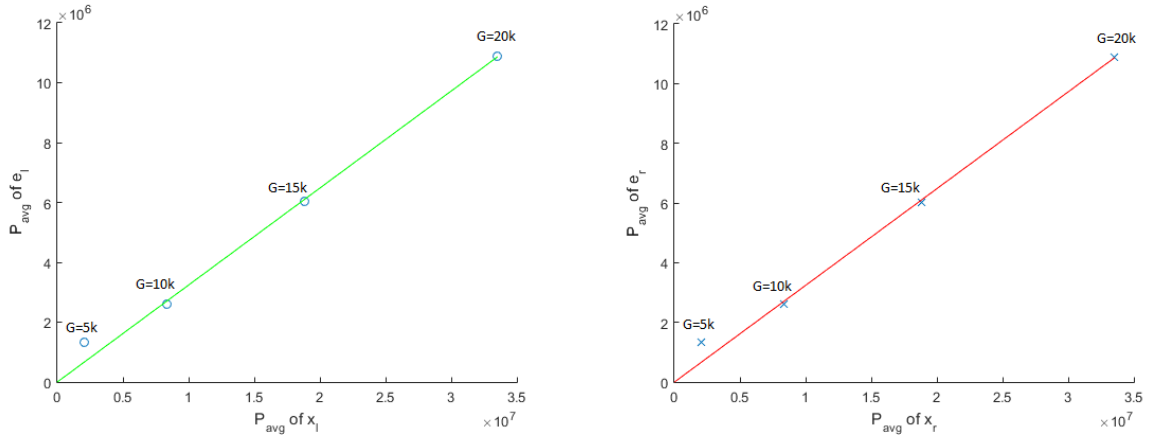
Non-linear distortion is a term used in the electronics and audio fields to describe the phenomenon of a non-linear relationship between the ”input” and ”output” signals of an electronic device. For this test, the linearity between the input power and the output power was studied

To find it, the secondary path was estimated with the off-line modelling algorithm. In each trial the training signal $x(n)$, white noise, had the digital gain Gd varied with the values of the set $Gd \in \{5000, 10000, 15000, 20000\}$. After each modelling run, the power of the computed error signal $e(n)$ and of the training signal was estimated with the squared average (the arithmetic mean of the squares of a set of numbers). If, with each variation of input power, the output power does not increase in the same way there are non-linearities present in the circuit.

As observed in Figure 5.4 there is a linear relationship between the input and output powers except for the lowest digital gain used, $Gd = 5k$, which exhibited high non linearity for low power signals. This result should be more linear.

5.1.3 Total Harmonic distortion

Total harmonic distortion (THD) is used to characterize the linearity of audio systems. In audio systems, lower distortion means the components in a loudspeaker, amplifier or microphone or other equipment produce accurate reproduction of audio. When a signal passes through a non-ideal, non-linear device, additional content is added at the harmonics of the original signal frequencies. A linear system with a



(a) Left Channel (b) Right Channel

Figure 5.4: Secondary path Input-Output linearity.

sinusoidal signal as input has always a sinusoidal signal with the same frequency at the output. THD is a measurement of the extent of non-linear distortion.

To determine the THD of a system for a given frequency, a sinusoidal signal with that frequency is put in the input a system. The output is acquired and the THD is calculated with

$$THD = \frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + \cdots + V_i^2}}{V_1} \tag{5.1}$$

where V_n is the RMS voltage of the ith harmonic and i = 1 is the fundamental frequency.

To evaluate the secondary path THD, a set of internally generated sinusoids with amplitude of half of the numerical range $A = 32767/2 \simeq 16384$ and frequencies $f \in \{50, 75, 100, 250, 500, 750, 1000\}Hz$, was sent to the loudspeaker. The data was acquired for an interval of 40 seconds, resulting in 640000 samples, to reduce the influence of the background noise. Then the THD was computed for each measurement.

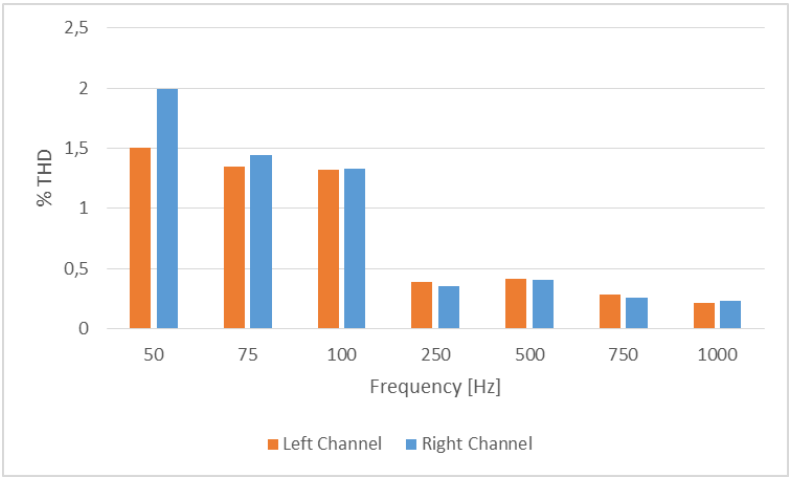


Figure 5.5: Secondary path total harmonic distortion.

In Figure 5.5, it can be observed that frequencies below or equal to 100 Hz have THD between 1%

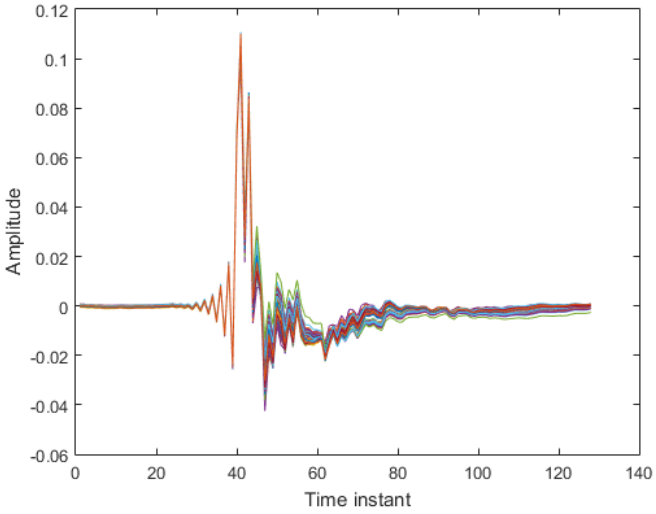


Figure 5.6: Estimated secondary path impulse responses.

and 2%. The rest of the frequencies have THD below 1%. The lower frequency results seem to be not very good (generally devices are rated below 1% THD), however 1% distortion will be only noticeable for noise cancelling levels close to 40 dB, which is a great deal more than what is expected for the attenuation performance of this system. Therefore the THD results are satisfactory. A source for those low frequency distortion may be from the electret microphones, since they are designed to work at higher frequencies [62, 63], and therefore may add distortion at low frequencies.

5.2 Secondary path off-line modelling

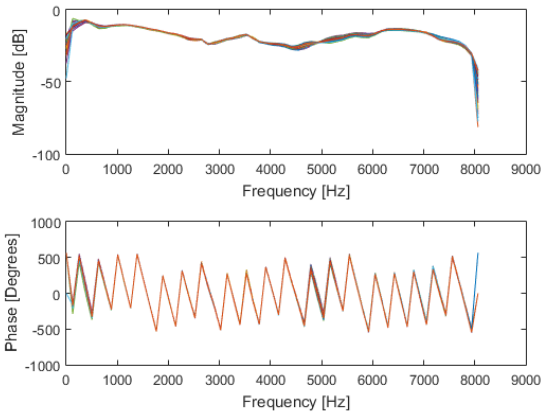
The off-line secondary path modelling is used for most of the algorithms. Thus, it is important to evaluate how the system performs when doing this procedure. For this measurement the right channel results are presented since the left channel results had great variation, probably due to a bad placement of the headset on the head for some measurements. The parameters for this experiment were:

- $\mu_s = 0.01$;
- $\lambda = 1$;
- Training signal maximum amplitude, or digital gain, $Gd = 19000$;
- $F_s = 16$ kHz;
- $L_s = 128$;
- $Ns = 5000$;
- $Gv = 0.4$
- $A_{V_{\text{pre}}} = 4$.

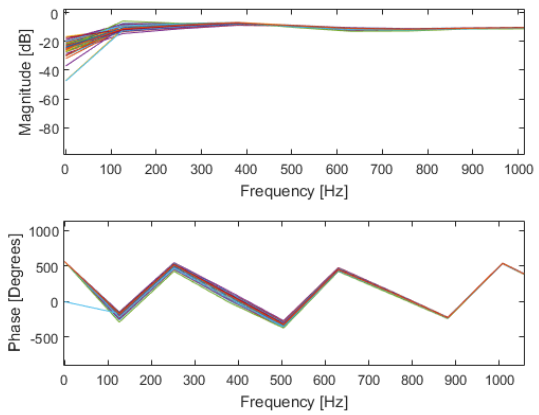
To assess the performance, 100 runs of off-line modelling were realized to visualize the variation in the impulse response and frequency response of the estimates. These do not have to exactly match the secondary path [3], but, as already stated, the phase error cannot be superior to $90°$. To compute it, one can assume that the variations of the estimates are due to slow changes on the secondary path, so

the maximum difference between the estimates phase is a good approximation of the phase error which an estimate may get.

Figure 5.6 shows the acquired impulse responses. Each line represents an estimate of the secondary path. It can be seen that there is great precision in modelling the delay of the secondary path, i.e. the large peak, because on that point the estimates are not spread. Thus, one can conclude that there is a big possibility of the algorithm accurately estimating the secondary path delay. However there is a big spread on the signal segment of the peak settling, which may turn out to be indicative of instability problems on the system.



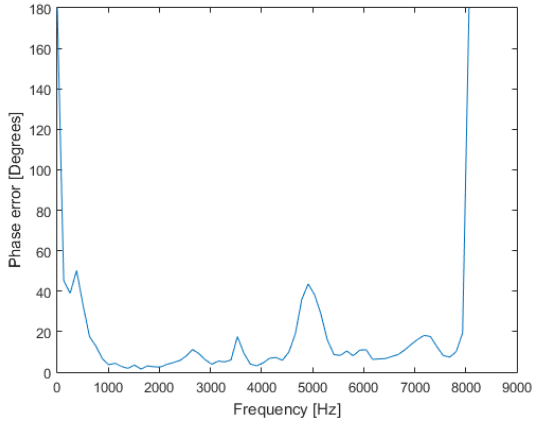
(a) Magnitude and Phase plot of the secondary path estimates.



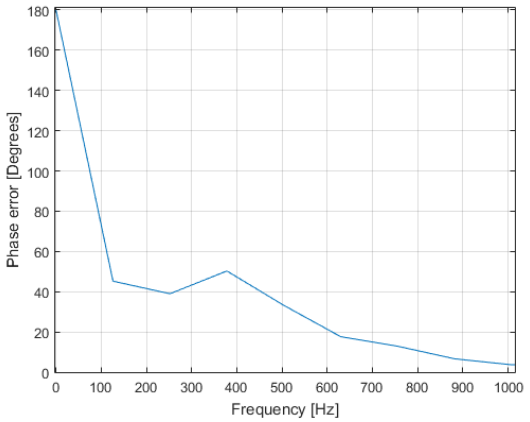
(b) Detail on the frequency response plot of the secondary path estimates.

Figure 5.7: Frequency response of the secondary path estimates.

With the acquired signals one can compute the frequency response of the estimates, seen in Figure 5.7. One can observe that the magnitude has a big variation on the band under 500 Hz and over 3500 Hz, specially under the 100 Hz. The phase also has a bigger variation on the lower end of the spectrum. This is better evaluated with a graph of the maximum difference between the estimates phase, represented in Figure 5.8.



(a) Maximum difference between the secondary path estimates phase .



(b) Detail on the phase error plot of the secondary path estimates.

Figure 5.8: Phase error of the secondary path estimates.

This shows that there is a large phase error at the frequencies where the secondary path has a small amplitude, as expected, due to the lower signal to noise ratio in these frequencies. Under approximately 80 Hz the phase error is above 90° thus, at these frequencies the system should be more prone to instability or at least slower convergence.

Note that the previous measurements were acquired at the time of development of the algorithms, thus the amplifier gains were not yet set, following the procedure presented in Chapter 4, nor the multi-rate processing was implemented, to reduce the sampling rate to 2kHz.

5.3 FxLMS feedback algorithm

In this section the FxLMS applied to a feedback system. The optimization process and performance assessment were started with this algorithm to be used as a comparative measure with the other algorithms and configurations, which are variations or slight changes on this one. Therefore, the statements made on chapter 2 will be possible to confirm in section 5.7.

5.3.1 Experimental optimization of the system's parameters

After adjusting the parameters of the designed hardware so that is working as expected, i.e. the gains of the headset amplifiers are set, the process of optimization may begin. In this procedure the system parameters are manipulated to achieve the best possible performance. The key parameters to be set are the value of the step size μ, the value of the leakage factor λ and the adaptive filters length L. Data is gathered while varying one of these variables and holding all others at a constant value to best isolate its effect on the system.

The experimental setup for the parameter optimization is depicted in Figure 5.9. This setup was used for all of the performance measurements, for the optimization and for the different primary noise, of all the used algorithms. Even though it depicts a feedforward system, when using feedback algorithms the reference microphone was turned off.

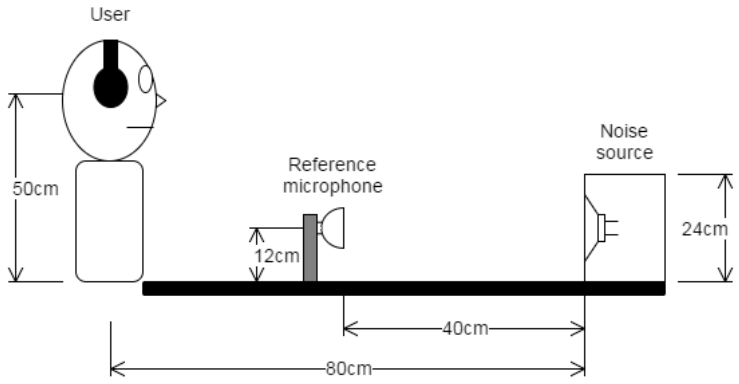


Figure 5.9: Experimental setup of the performance evaluation.

The noise source utilized was a Pioneer XR-NM1 stereo system. Its speakers have a 6Ω rated impedance and a frequency range of 55-20000 Hz [73]. It amplified a sinusoidal signal, created by a

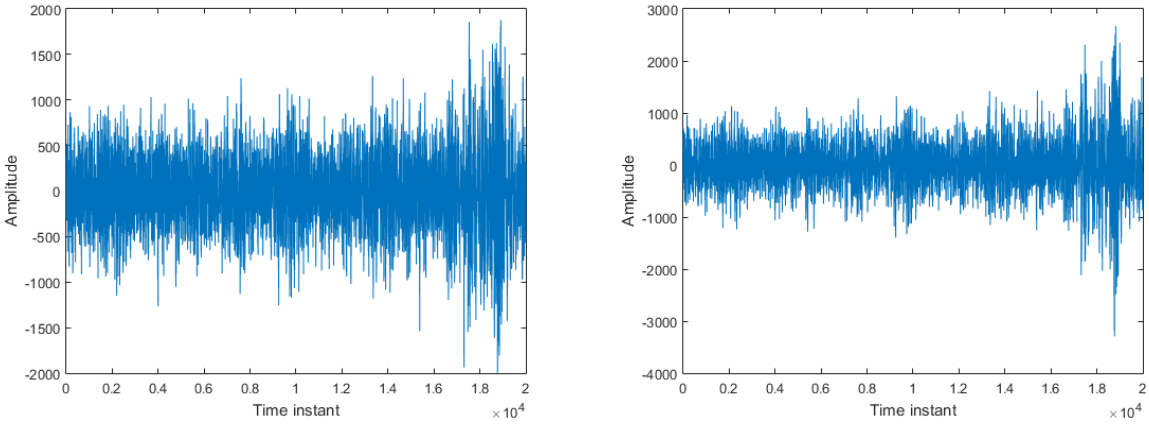
function generator, with 100 Hz, thus producing the primary noise used for the parameter optimization. This signal had two levels of sound pressure: a ”weak noise” with around 87.42 dB SPL, output by the ADC as a signal with maximum amplitude of 8000, equivalent to traffic on a busy road heard at 10 meters; a ”loud noise” with nearly 98 dB SPL, output by the ADC as a signal with maximum amplitude of 28000, equivalent to a noisy factory. However, during the off-line secondary path modelling the primary noise cannot have this sound pressure, or else the secondary path will have considerable errors thus making the system unstable.

It should be noted that the user was located in front of the noise source in all the trials, to maximize the sound pressure captured and balance the sound level for both channels. Also, the distances meet the causality constraint and increase correlation between the reference and error signal. However, the reference signal is not very good quality due to its distance to the noise source.

By default, the parameters used for this experiment were the ones that gave the best results after implementing the system. They were:

- Control filter NLMS step size $\mu = 0.001$;
- $\lambda = 1$;
- $\mu_s = 0.01$;
- $Gd = 10000$;
- $F_s = 2$ kHz;
- $W(z)$ adaptive filter length $L_w = L_s = 40$;
- $Ns = 4000$;
- $Gv = 0.12$;
- $A_{V_{\text{pre}}} = 55$;
- Reference microphone pre-amplifier gain $A_{V_{\text{ref}}} = 62.5$.

For this set of parameters, the ambient noise obtained, depicted for both channels in Figure 5.10, has a typical maximum amplitude of about 1000 or 60 dB. So, due to the numerical range being [-32768,32767], the maximum attenuation should be around 30 dB.



(a) Left Channel. (b) Right Channel.

Figure 5.10: Ambient noise at the time of the performance evaluation.

Table 5.1: Performance of FxLMS as a function of μ.

FxLMS_fb	
Step size	Att[dB]
0,0005	16,41
0,001	24,48
0,005	25,54
0,01	23,37 (unst)
0,05	18 (unst)

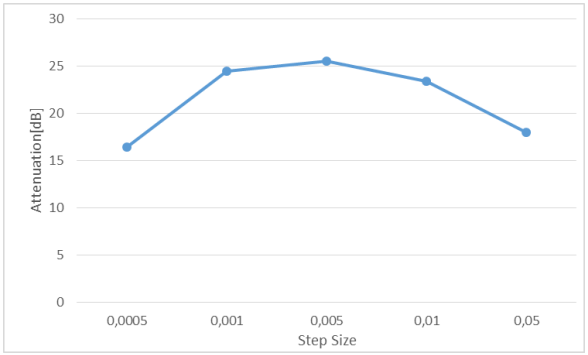


Figure 5.11: Noise reduction of FxLMS as a function of μ.

As stated above the left and right channel have similar characteristics. Their hardware is configured in the same way (preamp and power amp) and it is assumed the loudspeakers and microphone have similar electrical characteristics. Therefore the following sections will only illustrate the results for one of the channels.

The data acquisition ran for 10 seconds (or 20000 samples), 5 of which was spent with the ANCsystem off, sensing the primary noise, and the next 5 with the ANC system on, obtaining the error signal.

The following sections will repeat the steps stated in this one, so the procedure will be omitted unless there were changes to be noted.

Determining the value of the step-size

The step size μ was the first parameter that was altered to see how it affected the performance of the system. Each run of the noise cancelling algorithm used one of the step sizes in the set $\mu \in \{0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 1.5, 2\}$ to cancel the ”weak noise”. Note that since the adaptive algorithm used is the NLMS, the last step size is the upper limit of the NLMS stability bound. Then, the ratio between the average power of the last 2000 samples of the primary noise and of the residual error signal was computed, thus obtaining the an estimate of the attenuation. The convergence time was also determined, which was the time the system took to reach the maximum attenuation. Some observations on the stability were also made. The stability was categorized as:

•oscillating (osc) if the step size led the system to oscillations of the error signal in some of the runs but did not diverge;

•unstable (unst) if the algorithm diverged in a minority of the runs or if they converged but were prone to diverge with small, non-induced perturbations;

•very unstable (v.unst) if the instability was observed on a higher number runs;

•divergent if it simply diverged all the runs;

•stable if it did oscillate very few runs or did none of the above.

One thing that can be seen from Figure 5.11 is that the system did not have a linear dependence on the step size of the adaptive filter. As the value of μ increased from the low value of 0.0005, the performance of the system increased at first, then peaked at μ = 0.005. As represented in Table 5.1 μ =0.01 and μ =0.05 turns the system unstable and beyond that value the system always diverged, thus the performance of the system tended to decline beyond μ = 0.005. The convergence time followed

the same tendency, it reduced from over 5 seconds of the lowest step size until μ =0.01, with 0.35 seconds, then it increased at μ =0.05.

After the first set of trials, the three best step sizes that met the maximum attenuation, convergence speed and stability criteria, were chosen. They were once more compared with each other, this time by cancelling the ”loud noise”. The attenuation performance of this test ranked them from first to third to use them on the rest of the parameter optimizations. Figure 5.12, depicts the results of that test.

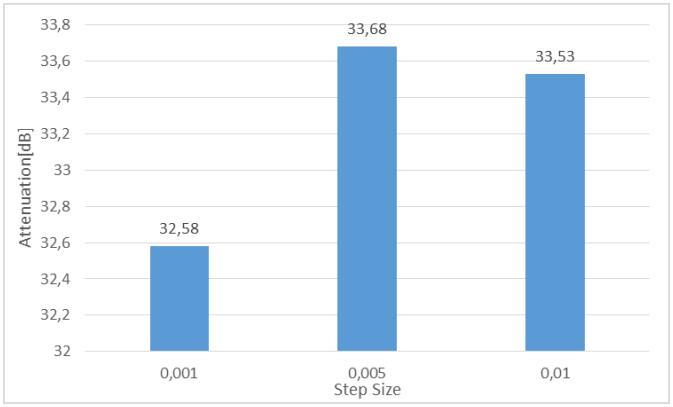


Figure 5.12: Performance of FxLMS with the loud noise.

The best attenuation and stability came from $\mu = 0.005$ however $\mu = 0.01$ got the best convergence speed. It should be noted that for these loud noises the stability of the system degrades a great a deal.

Determining the value of leakage factor

There is no reason to test the effect of the leakage factor for all of the values of μ. But one cannot simply use the best step size since there is nothing guaranteeing that it yields the best performance combined with a leakage factor. Also, it is still of interest to watch the behaviour of the performance for the top three while varying the rest of the parameters.

It was decided that the value of the leakage factor was to be varied in the set $\lambda \in \{0.99, 0.999, 1\}$. For each step size the leakage factor adopted one of the values mentioned. Table 5.2 shows the attenuation results and Figure 5.13.

Table 5.2: Performance of FxLMS as a function of λ.

FxLMS_fb		
μ	λ	Att[dB]
0,001	0,99	0,29
	0,999	2,87
	1	24,48
0,005	0,99	1,461
	0,999	10,98
	1	25,54
0.01	0,99	2,97
	0,999	15,03(osc)
	1	23,37(unst)

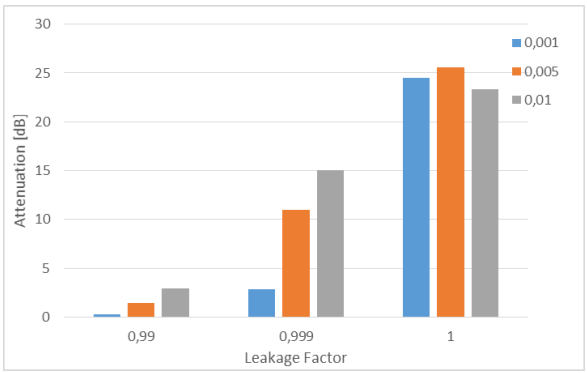


Figure 5.13: Noise reduction of FxLMS as a function of λ.

As seen, the leakage factor limits the attenuation of the system, with the $\lambda = 0.99$ being an un-

desirable value. Since the system has less to attenuate, the convergence time is greatly reduced as the leakage factor is reduced. On the other hand it improves the stability of the system. In cancellation attempts where the system would diverge with $\mu = 0.01$, the error signal now oscillates without diverging.

The usage of the leak factor was a nice cue to test the robustness of the system, since it is a method used to improve that characteristic. To do that, the situation when a user takes/puts the headset off/on, one of the most extreme perturbations, is performed. The robustness was improved because for $\lambda = 1$ the system always diverged when subject to great perturbations like the put on/take off situations; with $\lambda = 0.999$ the system does not always diverge, and when it does, specially for $\mu = 0.01$, it converges again maintaining the function.

The second best step size, $\mu = 0.01$, was put under test with the "loud noise". This time, to see if the stability improved without too much attenuation decay, using the leakage factor 0.999. The attenuation obtained was 18,38 dB and the stability did improve, however extreme perturbations still made the system diverge.

Determining the filters length

The last of the parameters that needed to be adjusted for this ANC system configuration was the filters length L. To reduce the amount of trials, the lengths of the adaptive filters $W(z)$ and $\hat{S}(z)$ were made the same. For the same reason, instead of using the top three step sizes, the second best $\mu = 0.01$ was again used in an attempt to improve the stability. This was of no problem, because if the second best step size improved performance the best step size would probably improve as well. The leakage factor used was $\lambda = 1$ because, outside of the slightly worse stability under extreme situations, it was clearly the best choice for this system.

First the lowest possible filter length, without making the system diverge, was determined. Then, the same was made to find the highest possible value. With these limits obtained, the interval was divided in around 7 or 8 tiers more or less with the same distance between them. The default filter length $L = 40$ was also included in the analysis.

The data is showed in Table 5.3 and graphically represented in Figure 5.14.

Table 5.3: Performance of FxLMS as a function of L.

FxLMS_fb	
L_w=L_s	Att[dB]
30	23,43 (osc)
40	23,37 (osc)
75	21,74
120	21,79
165	24,8 (osc)
210	27,21 (unst)
255	26,76 (unst)
300	21,82 (unst)

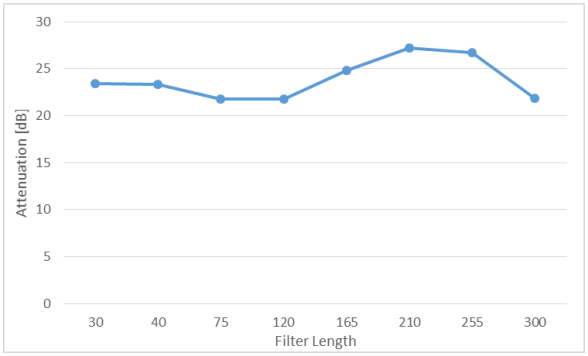


Figure 5.14: Noise reduction of FxLMS as a function of L.

For $L < 30$ the system diverges. For $L < 75$ the system oscillates and with $L = 75$ the system still oscillates, but less frequently and with smaller oscillations. The filter length $L = 120$ was the only value

that improved the stability and it was adopted for that reason. The instability observed for the rest of the values could be due to the length of the impulse response increasing the frequency resolution of the frequency response, thus the modelling error would have a stronger effect. The convergence time increases with the filter length as expected. The time spent by the processor in doing the calculations required by the ANC algorithm (processing time) for the maximum filter length tested was $t_p = 50\mu$s, a very small part of the sampling period ($T_s = 500\mu$s).

5.3.2 Noise reduction by primary noise

The system performance was optimal for the following set of parameters:

- $\mu = 0.005$;
- $\lambda = 1$;
- $L_w = L_s = 120$;

Step size $\mu = 0.005$ was revealed to be the best for the reasons stated and because $\mu = 0.01$ could not be stabilized and maintain attenuation for loud noises.

With the system calibrated, it was time to test the performance for various types of primary noise. The following tests used the same experimental setup of the previous subsection, as depicted in Figure 5.9. First, the sinusoidal signal, created by the function generator, had the frequency varied with the following values $f \in \{60, 100, 200, 400, 500\}$ Hz. The lower limit was determined by the noise source frequency range, and the higher limit by the band of interest considered, limited by the multi-rate filter passband. Then, a pre-recorded noise was fed to the noise source loudspeaker. These were an industrial compressor sound [74] for the narrowband noise, and a Brownian noise [75] for the broadband sound. This broadband noise was chosen because it has low frequency components with greater power.

To obtain the maximum possible attenuation, only loud noises were used. However, since the system was very unstable for the previous sound pressure, for these tests the sound pressure was reduced to nearly 96 dB SPL, output by the ADC as a signal with maximum amplitude of 22000, equivalent to a subway noise. Because of this it is expected for the maximum attenuation to decrease, compared to the same situations on the parameter optimization, since there is less power to attenuate and the bottom power, the ambient noise, stays the same.

The attenuation of the system was acquired by first using the error microphone to acquire the primary noise during 5 seconds, while the ANC was off, and then the error signal during another 5 seconds while the ANC was on. Then, an estimate of the attenuation was calculated in three ways; the power spectrum density of the last 2000 samples of both signals was obtained and the ratio of the average power of both signals was computed; the ratio of the average power of the last 2000 samples of both signals was computed. Finally, an average of the two methods was calculated.

Tone Frequency Sweep

Figure C.1 depicts the noise cancellation results obtained in real time for the tone frequency sweep. It is observed that the attenuation stays nearly even from 100 to 500 Hz. The lowest frequency noise had the worst attenuation but it was considerably distorted for 60 Hz, having harmonics also at 120 Hz,

180 Hz and 240 Hz which were all attenuated. The system had low stability since it was unstable for $f = 100$ Hz.

It should be noted that the system still cancelled for noises above the frequencies tested, however the passband of the multi-rate filter prevented the acquisition of the error signal and an estimate of the attenuation.

Narrowband noise

Figure C.2 shows considerable noise cancellation for some harmonic components. The noise cancellation levels at 82, 121, 176, 207 and 320 Hz are 11.06, 10.88, 9.38, 18.35 and 18.35 dB, respectively. A 5.42 dB total noise reduction was obtained. The system was oscillating.

The residual signal has more power than the primary noise, on the frequency band higher than 500 Hz.

Broadband noise

Looking at figure C.3 it can be seen that the system is unable to cancel considerably none of the harmonics. The estimated total attenuation was 0.17 dB, confirming what is seen.

5.3.3 Sampling Rate comparison

After obtaining the best set of parameters for the feedback FxLMS, the algorithm was used to compare performance of cancellation of the 100 Hz tone with different sampling rates. This test was not made with performance optimization in mind but to show the different performances obtained during the development of the project and explain the reasoning behind the choice of sampling rate.

The sampling frequencies were:

•$F_s = 16$ kHz, used in the beginning of implementation of the system;

•$F_s = 8$ kHz, the sampling rate used on the first try of multi-rate processing, downsampled from $F_s = 48$ kHz;

•$F_s = 2$ kHz, the current sampling rate, downsampled from $F_s = 16$ kHz.

As seen in Figure 5.15, in terms of attenuation, the best sampling rate was $F_s = 8$ kHz and then $F_s = 16$ kHz. However, for $F_s = 16$ kHz the system revealed to be unstable for the step size used, needing a smaller step size to be stable and turning the system slower to converge. The best sampling rate turned out to be equally stable as $F_s = 2$ kHz however for other algorithms, specifically the BMFxLMS, it was less stable.

Another important fact to note, was that the robustness of the system improved for the lowest sampling rate because the system at 8 kHz could not handle perturbations such as strikes on the ear cups or pulls on the wires, which at 2 kHz is possible to do.

Finally, the smaller sampling rate allowed for usage of longer filters before the processing time gets higher than the sampling period. It also allowed for the use of shorter filters without the system diverging. However, $F_s = 8$ kHz increased the maximum filter length without the system diverging, to nearly 500 taps.

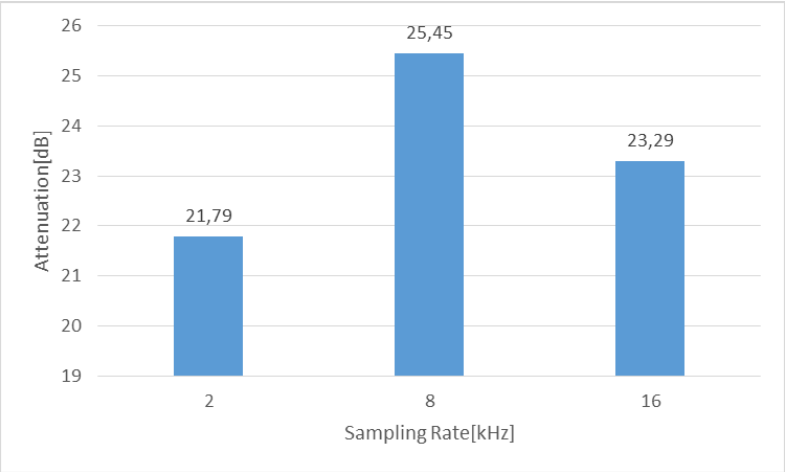


Figure 5.15: Noise reduction of feedback FxLMS as a function of F_s.

Therefore, another viable sampling rate would be $F_s = 8$ kHz but the efforts to optimize the stability and robustness would be greater.

5.4 BMFxLMS algorithm

The next algorithm used on the feedback system was the band limited MFxLMS (BMFxLMS). It was used in an attempt to improve the stability of the feedback FxLMS. It does this with a method to counteract the phase error typical of the off-line secondary path modelling.

5.4.1 Experimental optimization of the system's parameters

As stated, the procedure described in the previous section is repeated for the parameter optimization of this algorithm. The default parameter values indicated before are again the same.

Table 5.4: Performance of the BMFxLMS as a function of μ.

BMFxLMS_fb	
Step Size	Att[dB]
0,0005	3,697 (unst)
0,001	23,53
0,005	24,47 (osc)
0,01	23,35 (osc)
0,05	22,09 (unst)
0,1	16,84 (unst)
0,5	9.839 (v.unst)
1	23,88 (v.unst)
1,5	21,47 (v.unst)

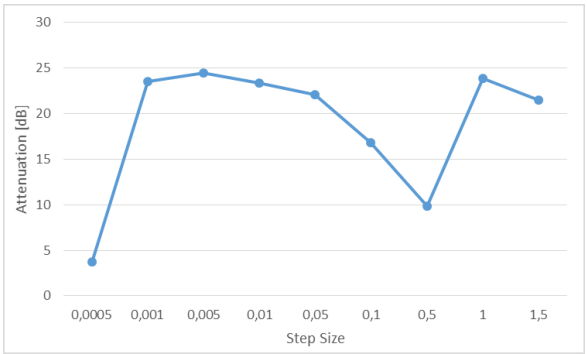


Figure 5.16: Noise reduction of BMFxLMS as a Function of μ.

Determining the value of the step-size

For this test, the system oscillated in almost all the trials. As will be seen at the end of this subsection, this effect will the reduced with a higher filter length. In any case, just by taking into account the relative

stability and the attenuation, seen in Figure 5.16 and Table 5.4, one can determine what are the top three step sizes, $\mu = \{0.001, 0.005, 0.01\}$. For the value $\mu = 2$ the system diverged.

For step sizes lower than 0.001 the convergence time was higher than 5 seconds. Then, until $\mu = 0.01$, the convergence time decreased with each increment. For step sizes higher than 0.01, the convergence behaviour was different from the usual exponential decay. In this situation, the convergence time is very short (under 0.1 seconds), however there was an oscillation that increases the time to nearly half a second.

The algorithm, using the top three step sizes, was subject to the ”loud noise”. As seen in Figure 5.17 the best step size is $\mu = 0.01$.

These results show the importance of selecting the top three step sizes based on the maximum cancellation, since the order from first to third best, based on the attenuation, obtained in the ”weak noise” experiment was changed.

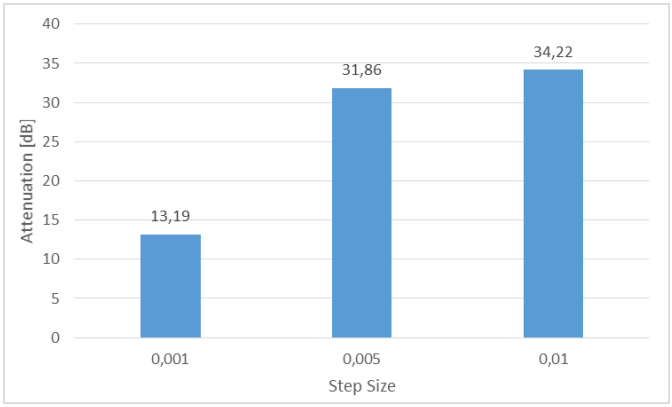


Figure 5.17: Performance of BMFxLMS with the loud noise.

Determining the value of leakage factor

Figure 5.18 and Table 5.5 once again confirm that the usage of a leakage factor reduces the attenuation, meaning that it really is comparable as adding white noise to the input of the adaptive filter. The smallest leakage factor performed, once again, badly under the attenuation criterion.

Table 5.5: Performance of the BMFxLMS as a function of λ.

BMFxLMS_fb

μ	λ	Att[dB]
0.001	0,99	0,6
	0,999	2,678
	1	23,53
0.005	0,99	1,31
	0,999	8,231
	1	24,47 (osc)
0.01	0,99	1,9
	0,999	10,05
	1	23,35 (osc)

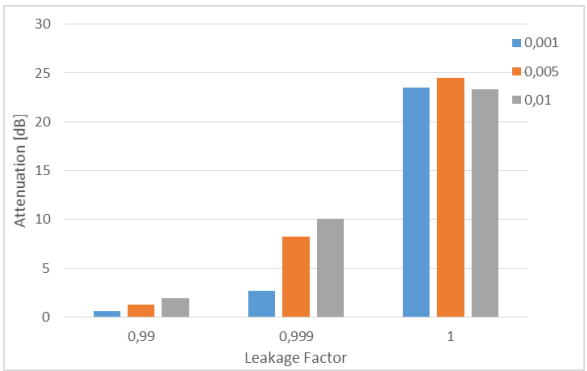


Figure 5.18: Noise reduction of BMFxLMS as a function of λ.

The robustness was tested during the leakage factor optimization. With $\lambda = 1$, the system diverges

and recuperates function for the two best step sizes $\mu = 0.005$ and $\mu = 0.01$, during the put on/take off perturbation. For $\lambda = 0.999$, the robustness was improved since the system did not diverge with extreme perturbations. The convergence speed and stability are improved as the leakage factor is decreased.

In [23] it is recommended to use $\lambda \neq 1$. However, this was not followed because the attenuation would drop, even though robustness and stability would suffer from that.

The combination of the best step size, $\mu = 0.01$, and $\lambda = 0.999$ was tested with the "loud noise". The robustness was improved because the system would rapidly recover from divergence induced by extreme perturbations. However, this came at the cost of reduced attenuation, achieving only 12,57 dB.

Determining the filters length

To determine the best filter order for the adaptive filters, the best step size, $\mu = 0.005$ was used. The leakage factor used was $\lambda = 1$, as attenuation was preferred to stability. As seen in Table 5.6 and Figure 5.19, the stable limits of the filter length were $L = 30$ and $L = 80$. Below and above these values the system diverges.

Table 5.6: Performance of the BMFxLMS as a function of L.

BMFxLMS_fb	
$L_w = L_s$	Att[dB]
30	24,92 (osc)
40	20,11 (osc)
50	25,18
55	26,71
60	23,66
70	15,72
80	11,56 (osc)

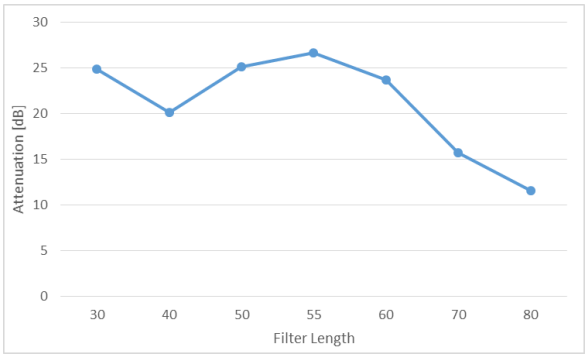


Figure 5.19: Noise reduction of BMFxLMS as a function of L.

The best performance was achieved when the length of the filter was $L = 55$. The performance starts to drop for $L > 60$. As the filter length was increased, from the value used on the previous tests, $L = 40$, the stability greatly improved. The convergence time increased with the filter length and the processing time of the maximum filter length tested was $t_p = 14\mu$s.

5.4.2 Noise reduction by primary noise

The system performance was optimal for the following set of parameters:

- $\mu = 0.005$;
- $\lambda = 1$;
- $L_w = L_s = 55$;

The leakage factor used was $\lambda = 1$ for the preference of attenuation to stability and robustness. If the system proved to be unstable for loud noises, one could always change it.

The test conditions are maintained from the previous section.

Tone Frequency Sweep

In Figure C.4 it is observed that the attenuation increased with the frequency.

It should be noted that the attenuation for a 100 Hz tone is so low because the system oscillated with loud noise.

Narrowband noise

The BMFxLMS algorithm attenuates a small number harmonics of the narrowband noise, as seen in C.5. The frequencies showing large attenuation are 121, 176, 207 and 480 Hz are 9.35, 6.18, 8.12 and 7.86 dB, respectively. A 1.13 dB total noise reduction was obtained. The system was stable during this test.

Broadband noise

The broadband noise, once again was not cancelled, as expected. Figure C.6 depicts this experiment results. The total noise reduction was 0.98 dB, confirming what was seen.

5.5 FxLMS feedforward algorithm

This is the FxLMS algorithm applied to a feedforward configuration of the system. It was used to demonstrate the better performance of feedforward over feedback configurations. It also serves as a standard of comparison between the FxLMS and the on-line modelling variation MMFxLMS.

5.5.1 Experimental optimization of the system's parameters

For this algorithm, the performance of the system was analysed for an additional parameter variation. This was the gain of the reference microphone pre-amplifier, needed for feedforward algorithms. The power of the reference signal had weight on the misadjustment, as stated in chapter 2, and therefore the maximum attenuation of the system.

The parameters used in this set of tests are the same ones used in subsection 5.3.1, with an addition of the default value of the reference microphone pre-amplifier being $A_{V_{\text{ref}}} = 62.5$. The "loud noise" sound pressure was reduced to the level used in subsection 5.3.2, because the system was unable to cancel the regular sound pressure.

Determining the value of the step-size

In Table 5.7 shows all the results for the optimization of the step-size for the feedforward FxLMS and Figure 5.20 illustrates it in a graph.

The attenuation increases from 0.0005 until it reaches a peak, now at 0.01. From there on, the performance degrades and the system diverges at $\mu = 0.1$. This algorithm was stable for step size values lower than 0.05. Note that the attenuation would be better if the default reference signal power was set to the lowest possible level, as described in the end of the subsection.

The convergence time decreased while the step size increased, until it got to a minimum at 0.01, then it increased again due to the instability of 0.05. The lowest step size converged slightly under 5 seconds and the lower values over 5 seconds.

Table 5.7: Performance of feedforward FxLMS as a function of μ.

FxLMS_ff	
Step Size	Att[dB]
0,0005	13,56
0,001	14,53
0,005	19,31
0,01	21,68
0,05	21,50(unst)

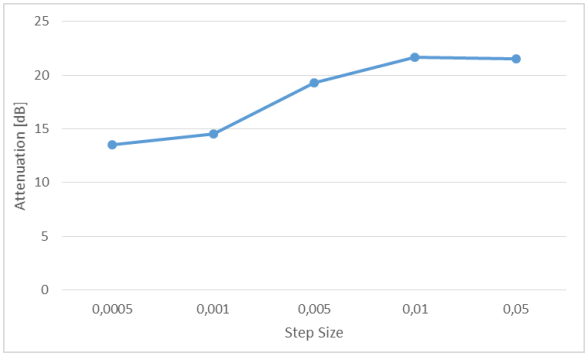


Figure 5.20: Noise reduction of feedforward FxLMS as a function of μ.

The three best step sizes, 0.005, 0.01 and 0.05, were used to cancel the ”loud noise”. Figure 5.21 depicts that, once again, the best of the three changed from the ”weak noise” case, being $\mu = 0.005$. The stability did not degrade considerably with the use of the ”loud noise”.

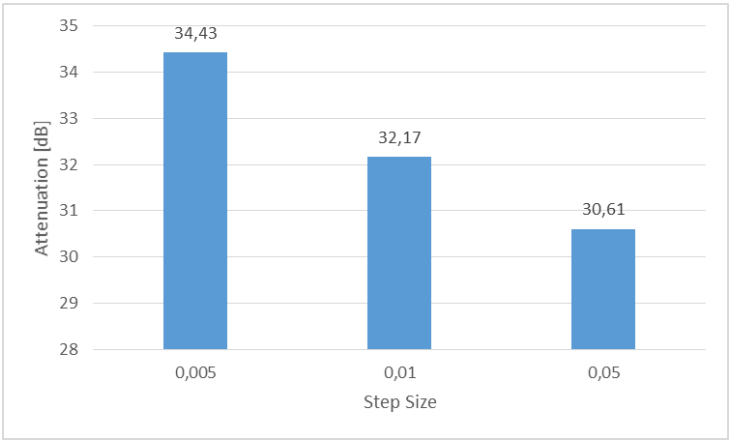


Figure 5.21: Performance of feedforward FxLMS with the loud noise.

Determining the value of leakage factor

For the following experiment, with results presented in 5.8 and illustrated in 5.22, the behaviour of the system was not the expected since the attenuation got better for one of the cases, and there was not an improvement of stability nor robustness.

This time the attenuation improved with the addition of the leakage factor $\lambda = 0.999$, for $\mu = 0.05$. This confirms the assumption that the best step size does not yield the best results. However the stability got degraded for the same leakage factor since the algorithm diverged one out of three runs. The robustness was not improved as well because only for the combination of $\mu = 0.005$ and $\lambda = 0.999$ the system would recover after diverging when subject to great perturbations. The smallest leakage factor, 0.99, still worsens attenuation a great deal. However, the usage of this value could be justified by the improved robustness, since it did not diverge at all for great perturbations.

The time of convergence was tied to the stability of the system. For $\lambda = 0.999$ the convergence time increased comparing to $\lambda = 1$ but the smallest leakage factor reduced the convergence time a great

Table 5.8: Performance of feedforward FxLMS as a function of λ.

FxLMS_ff

μ	λ	Att[dB]
0.005	0,99	1,31
	0,999	8,931 (unst)
	1	19,31
0.01	0,99	2,888
	0,999	16,68 (unst)
	1	21,68
0.05	0,99	8,225
	0,999	23,44 (unst)
	1	21,50 (unst)

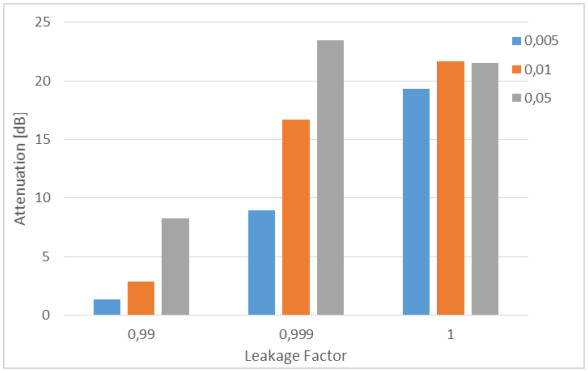


Figure 5.22: Noise reduction of feedforward FxLMS as a function of λ.

deal.

For the loud noise test, one should have used the best combination of values. However, due to the instability obtained, the best step size $\mu = 0.005$ was used instead, with $\lambda = 0.999$. Contrary to the weak noise cancellation, the system was not unstable and attenuated 13.27 dB. When the robustness got tested, the algorithm did not diverge.

Determining the gain of the reference pre-amplifier

For this procedure, the best step size $\mu = 0.005$ was employed instead of using the combination that gave the best attenuation with this system arrangement, $\mu = 0.05$ and $\lambda = 0.999$. To be a more complete study, the top three step sizes should be used.

Three pre-amplifier gains were selected to use in this study, to ease the adjustment of the potentiometer: a minimum value with $A_{V_{\text{ref}}} = 6.25 = 15.92$ dB; a medium value with $A_{V_{\text{ref}}} = 22.5 = 27.04$ dB; a maximum value $A_{V_{\text{ref}}} = 62,5 = 35.92$ dB.

The best value for the microphone pre-amplifier was found to be the lowest setting of 15.92 dB, as shown from the data graphed in Figure 5.23, because it had the highest attenuation.

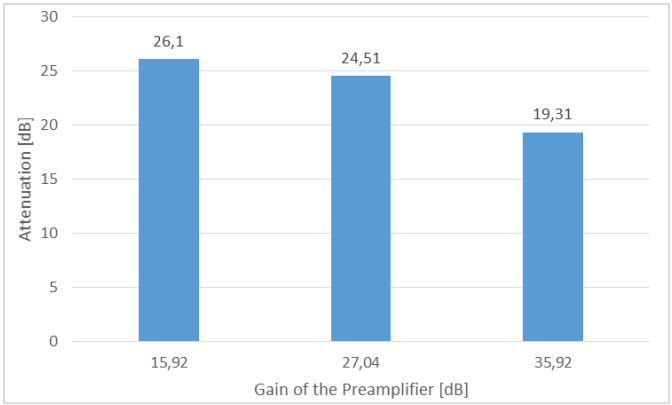


Figure 5.23: Noise reduction of the feedforward FxLMS as a function of pre-amplifier gain.

Unfortunately, this data was processed only after the following experiments were already completed. Also, the differences in attenuation could not be discerned just by looking at the error signals. Because of that, the pre-amplifier gain used for the rest of the chapter was the default, the maximum gain.

Determining the filters order

The results of the filter length variation are shown in Table 5.9 and graphically represented in Figure 5.24. The step size used was $\mu = 0.005$ and the leakage factor $\lambda = 1$. The maximum pre-amplifier gain was still used for the reasons stated previously.

Table 5.9: Performance of the feedforwad FxLMS as a function of L.

FxLMS_ff	
$L_w = L_s$	Att[dB]
8	23,74
40	19,31
200	25,62
300	23,75
500	21,57
600	18,77
800	25,59 (unst)

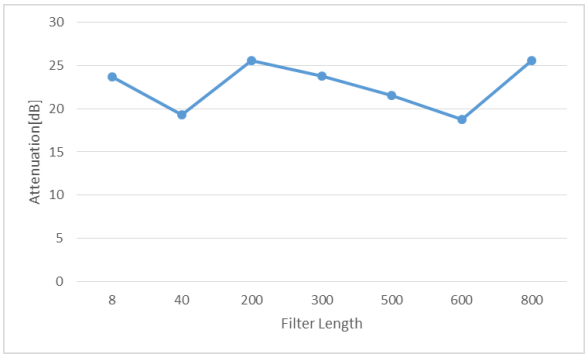


Figure 5.24: Noise reduction of feedforwad FxLMS as a function of L.

The stable limits of the variation were $L = 8$ and $L = 800$. Outside of this interval, the system always diverged. The best attenuation was observed at $L = 200$ and $L = 800$. However, to save on memory requirements and because of the better stability, the lower number of taps was selected as the best of the two. The system was not stable in all of the values in the interval. For instance, $L = 100, 400, 700$ made the system diverge. The convergence time also increased with the increments on the filter length. The maximum processing time was $t_p = 140\mu$s.

5.5.2 Noise reduction by primary noise

With the parameter optimization completed, a good set of parameters was:

- $\mu = 0.005$;
- $\lambda = 1$;
- $L_w = L_s = 200$;
- $A_{V_{\text{ref}}} = 62.5$;

As stated these cannot be called the optimal ones since the other possibilities found were not pursued. In any case the attenuation, stability and convergence speed were reasonable enough to proceed the algorithm performance analysis.

The procedure and experimental setup was maintained from subsection 5.3.2.

Tone Frequency Sweep

The feedforward FxLMS increases the attenuation with frequency except for the noise with 200 Hz, as depicted in Figure C.7. This drop in attenuation is due to the fact the system oscillates, even though very little, for that frequency.

Narrowband noise

One can see two bands in the power spectrum, represented Figure C.8, where the attenuation of harmonics was greater. The lower band, from 82 to 207 Hz, with less attenuation, and the higher band from 304 until 566 Hz with greater attenuation. Looking at the harmonics stated before, at 82, 121, 176, 207 and 320 Hz are 20.45, 19.62, 8.74, 18.63 and 22.29 dB, respectively. The total noise reduction was 7.11 dB and the system was very stable and resisted small perturbations, such as strikes in the ear cup or tugs in the wires. Also, there are few frequencies where the error signal has more power than the primary noise.

Broadband noise

To cancel broadband noise it is necessary to give a good reference signal to the system, in advance. However, as seen in Figure C.9, the feedforward signal could not cancel the brownian noise. This is further confirmed by the total attenuation achieved, -0.72 dB, meaning the error noise has more spectral power than the primary noise.

5.6 MMFxLMS algorithm

This algorithm is used in an attempt to improve the stability of the FxLMS. It does this with an on-line modelling algorithm which constantly updates the secondary path estimate, thus mitigating the phase error effects on stability.

5.6.1 Experimental optimization of the system's parameters

The default parameters used in subsection 5.5.1 and subsection 5.3.1 are once more used. The procedure will be the same, however this algorithm has the tendency to have overshoot at the beginning of the cancellation procedure, which will be accounted for the performance analysis. When there is an oscillation of the error signal to values above the maximum amplitude of the primary noise, overshoot is present and is measured as a ratio of the maximum amplitude of the signal.

Determining the value of the step-size

Table 5.10 has the data recorded for this set of measurements. Figure 5.25 illustrates the attenuation variation as the step size varied.

The algorithm always converges and the system is very stable. It only diverges for step values above 2, thus obeying the stability condition of the NLMS algorithm. The convergence time is reduced with each step size, from the minimum 5 seconds at $\mu = 0.0005$ to 0.25 seconds at $\mu = 0.5$. From then on the convergence time increases slightly. The overshoot increases with the step size and it increased much for step sizes above $\mu = 0.5$.

To cancel the "loud noise" used in subsection 5.5.1, the three best step sizes used were 0.1, 0.5 and 1 because they have a mix of stability, speed and attenuation that puts them above the rest, even though the overshoot is higher. In Figure 5.26 it is seen that the best step size in terms of attenuation is $\mu = 1$. The stability was not degraded with the use of high noise levels.

Table 5.10: Performance of MMFxLMS as a function of μ.

MMFxLMS_ff

Step Size	Att[dB]	Ovs[dB]
0,0005	15,22(osc)	-
0,001	24,75	-
0,005	17,99	-
0,01	24,94	-
0,05	20,91	-
0,1	23,22	0,7
0,5	26,09	0,9
1	22,43	4,59
1,5	21,84	3
2	22,8 (osc)	4,9

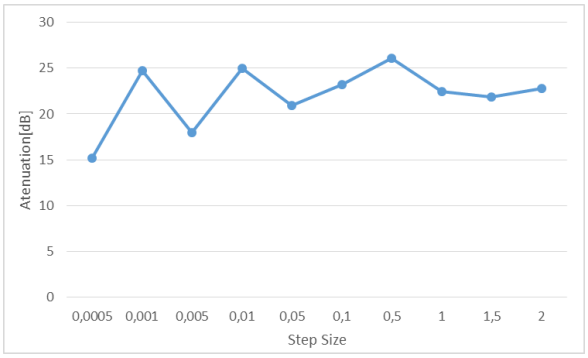


Figure 5.25: Noise reduction of MMFxLMS as a function of μ.

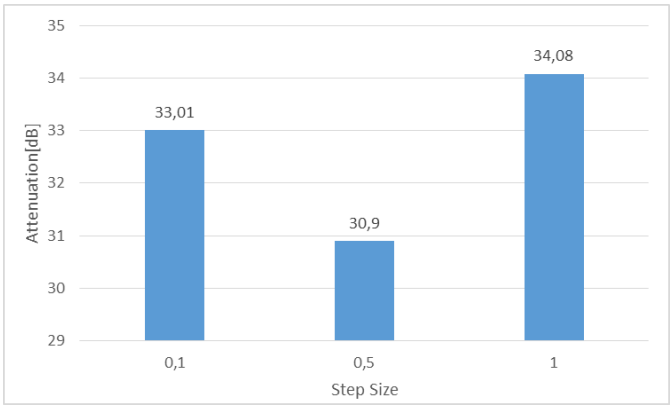


Figure 5.26: Performance of feedforward MMFxLMS with the loud noise.

The use of the stated noise level is justified as a preventive measure against the saturation of the channel, because the overshoot could exceed the numerical range of the CODEC.

Determining the value of leakage factor

With the three best step sizes selected, the performance of the system with the variation of the leakage factor was studied. Its results are shown in Table 5.11 and graphically represented in Figure 5.27.

Table 5.11: Performance of MMFxLMS as a function of λ.

MMFxLMS_ff

μ	λ	Att[dB]	Ovs[dB]
0,1	0,99	13,58	-
	0,999	22,77	-
	1	23,22	0,7
0,5	0,99	21,71	0,75
	0,999	22,74	2,754
	1	26,09	0,9
1	0,99	19,6	5,113
	0,999	28,42	4,74
	1	22,43	4,59

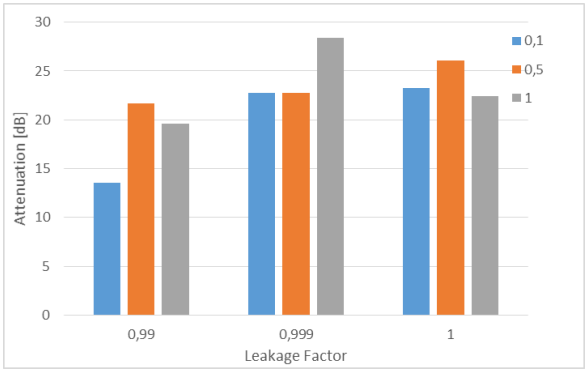


Figure 5.27: Noise reduction of MMFxLMS as a function of λ.

There was little degradation of the attenuation with the use of $\lambda = 0.999$. For $\mu = 1$ there was

an improvement of attenuation with that leakage factor. The smallest leakage factor continued to give the worse results. The system kept being very stable, with no oscillations, however robustness did not improve. This was because the system can only hold against small perturbations, such as pulls of the wire or touches on the headset, but, unexpectedly, it always diverged with the put on/put off situation. This could be due to the captured primary noise, not attenuated by the seal of the ears and ear cups, exceeding the numerical range of the CODEC. The convergence time stayed nearly the same for all leakage factors. The overshoot did not follow an apparent rule, since it increased, with the decrease of leakage factor, for some step sizes and decreased for others.

Testing the best step size and $\lambda = 0.999$ with the loud noise yield a slight improvement in attenuation, 34.17 dB with no decrease in stability and robustness. However the system continued to diverge when subject to extreme perturbations. The overshoot did not increase.

Since it was not apparent the improvement of performance before the processing of the data, the following experiments were all realized with $\lambda = 1$.

Determining the gain of the reference pre-amplifier

Following the procedure stated in the previous section and with $\mu = 1$ and $\lambda = 1$, the results of the attenuation performance with the variation of the reference microphone pre-amplifier gain are observed in Figure 5.28. The results are similar since the best gain still is the one that gives a lower power reference noise. However, for the same reasons stated in the previous section the maximum power was set instead.

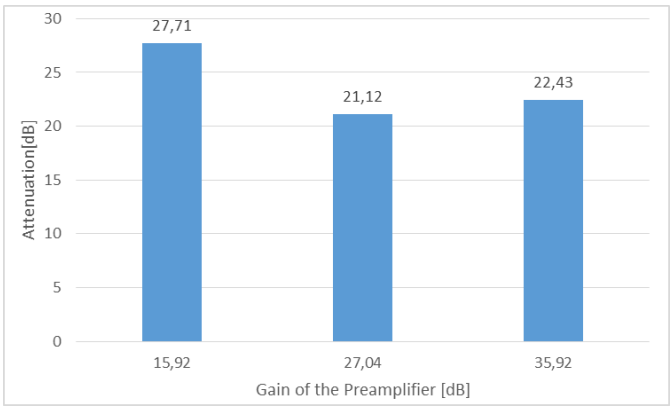


Figure 5.28: Noise reduction of the MMFxLMS as a function of the pre-amplifier gain.

Determining the filters order

This set of experiments is a statement of the stability of the algorithm. The results are shown in Table 5.12 and the attenuation variation depicted in Figure 5.29.

The limits of the interval where the system does not diverge are $L = 3$ and $L = 1300$. The best attenuation is observed at $L = 1200$. However $L = 200$ was chosen as the best filter length because the convergence time was smaller, there were memory savings to be made with it and the difference in attenuation was not noticeable, at least before the post processing of the data. The overshoot was found to be relatively the same with increasing length, not taking high values nor completely disappearing. The

Table 5.12: Performance of the MM-FxLMS as a function of L.

MMFxLMS_ff	
$L_w = L_s$	Att[dB]
3	22,83
40	22,43
200	26,3
400	22,29
600	24,91
800	26,54
1000	24,2
1200	28,41
1300	17,4

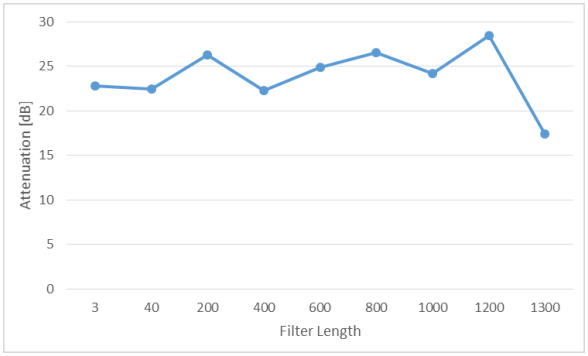


Figure 5.29: Noise reduction of MMFxLMS as a function of L.

convergence time increased with the length size, once again. The processing time for $L = 1300$ was $t_p = 490\mu$s, meaning that the algorithm is only unstable if the processing time exceeded the sampling period of 500μs.

5.6.2 Noise reduction by primary noise

It was found that a good set of parameters was:

- $\mu = 1$;
- $\lambda = 1$;
- $L_w = L_s = 200$;
- $A_{V_{\text{ref}}} = 62.5$;

Even though this was not the optimal combination of parameters, they proved to be reasonable to test with other primary noises.

The overshoot was revealed to be a problem when acquiring data because sometimes it was so high it saturated the CODEC. Therefore the procedure had to be slightly changed. Whenever the overshoot made the system diverge, first the overshoot was captured with very low noise levels, so it did not exceed the numerical range. Then, to obtain an attenuation value, the noise levels were decreased during the beginning of the ANC and were instantly increased after the overshoot passed.

Tone Frequency Sweep

In this experiment it was revealed that the overshoot was a crippling factor for the performance. The results can be seen in Table C.1 and the attenuation is graphically represented in Figure C.10. This time, the set of frequencies used is $f \in \{60, 80, 100, 200, 400, 500\}$ Hz. The best attenuation was observed in the band between 80 and 200 Hz. However the overshoot was very high for 60, 200 and 500 Hz. Below 80 Hz the system always had excessive overshoot. The step size was varied to check if it was a problem of optimization, however the overshoot levels were maintained.

Narrowband noise

The harmonics observed until now 82, 121, 176, 207 and 320 Hz had 23.41, 20.02, 8, 15.13 and 21.13 dB attenuation, respectively, as seen in Figure C.11. The total attenuation was 1.07 dB. Even

though it achieved considerable attenuation for the harmonics mentioned, the ANC system introduced more power for other frequencies.

For this noise, the overshoot did not saturate the system, however it did slow down the convergence of the system, since nearly half of the captured data were overshoot samples.

Broadband noise

In Figure C.12 it can be seen that, although the MMFxLMS algorithm attenuates for some frequencies, there are whole bands of frequencies that the system adds power to the system. This is confirmed by the total attenuation of the system -3.47 dB, meaning it adds noise to the system.

5.7 Discussion on results of the tests

In this section, the results of the performance tests made with varying primary noises will be discussed and the algorithms performance compared. Then the feedback FxLMS will be compared with the results from [51], [76] and [77].

Tone Frequency Sweep

The attenuation results of the tone sweep experiment, for each algorithm, are collected in Table 5.13.

Table 5.13: Attenuation of the algorithms with the tone frequency sweep.

Frequency [Hz]	FxLMS_fb-Att[dB]	BMFxLMS-Att[dB]	FxLMS_ff-Att[dB]	MMFxLMS-Att[dB]
60	21,62	23,73	19,62	16,62
100	28,83	18,94	26,69	29,6
200	28,27	28,45	20,52	29,72
400	26,58	24,9	28,23	23,77
500	26,35	38,93	32,37	25,03

On average, the best attenuation is from the BMFxLMS and the worse from the MMFxLMS. It should be noted that a lower reference signal gain the feedforward algorithms would have resulted in better attenuation. The feedback FxLMS algorithm was the least stable and the MMFxLMS was the most stable, since it only diverged when the overshoot exceeded the numerical range. Even then the system would recover and cancel. From these results, it can be concluded that the methods to counter the modelling error effects were successful in doing so.

In [51] the attenuation achieved for the same band was between 70 to 85 dB of attenuation. The discrepancy between the obtained value and these values can be explained by the usage in [51] of a microphone inside a head simulator and torso simulator to capture the residual acoustic pressure and the primary noise. This way the attenuation estimate would be better and would not include the electrical noise of the hardware. The system used in [77] had worse performance for the low frequency band since for $f < 100$ Hz had attenuation under 10 dB. But for frequencies between 200-500 Hz it had attenuation between 40 and 50 dB. In the same document, a fixed controller headset was tested and revealed to be inferior with attenuations under 10 dB.

Narrowband noise

In Table 5.14, the attenuation on the observed harmonics for each algorithm are compiled.

Table 5.14: Attenuation of the algorithms with the narrowband noise.

Frequency [Hz]	FxLMS_fb-Att[dB]	BMFxLMS-Att[dB]	FxLMS_ff-Att[dB]	MMFxLMS-Att[dB]
82	11,06	3,65	20.45	23.41
121	10,88	9,35	19.62	20.02
175,8	9,38	6,18	8.74	8
207	18,35	8,12	18.63	15.13
320	18,35	-	22.29	21.13
Total Attenuation	5.42	1.13	7.1	1.07

The feedforward FxLMS had the best total attenuation and on average of the observed harmonics. MMFxLMS had equally good average attenuation for those harmonics, however the overall performance was not very good. Although the feedback FxLMS had the second best total attenuation, it was the algorithm with the worst stability. The BMFxLMS improved on it and the feedforward algorithms had no stability issues. It can be seen that to improve the stability the attenuation was lowered as an effect and other unwanted effects appeared.

The propeller plane noise used in [76] as a narrowband sound had a power spectrum similar to the one used in this work. Even though it had better total attenuation, 15.1 dB, and better attenuation for 80 Hz, 26 dB, this system has better attenuation at frequencies higher than 100 Hz. The results obtained were attenuation levels of 26 dB at 80 Hz, 7 dB at 160 Hz, 14 dB at 240 Hz, 13 dB at 320 Hz and 6 dB at 400 Hz. The commercial headset systems tested in that work had attenuations comparable with the ones of this system. They were attenuation levels of 14 dB at 80 Hz, 16 dB at 160 Hz, 15 dB at 240 Hz, 12 dB at 320 Hz and 7 dB at 400 Hz and total attenuation of 12.8 dB for system-1. For system-2 were 15 dB at 80 Hz, 16 dB at 160 Hz, 6dB at 240 Hz, 0 dB at 320 Hz and -4 dB at 400 Hz with total attenuation of 9.6 dB.

The engine noise used in [51] had harmonics attenuated at 20.906, 18.387, and 16.293 dB for harmonics at 61, 122, and 183 Hz, respectively. However the results for frequencies higher than those were worse than those of the present system. The noise cancellation levels using the commercial headphone studied in that work for those three harmonics were 0.685, 8.284, and 14.575 dB, with an overall performance similar to the one obtained in this work.

Broadband Noise

Unfortunately, broadband noise was not cancelled by this system. For feedback systems that was expected, but for feedforward systems the physical placement of the sensors probably did not meet the causality and coherence conditions. The causality condition was not met probably due to the delays of the anti-aliasing and reconstruction filters of the CODEC being too large. It would be interesting to measure these filter delays and compare to the acoustic delay of the system. That could also have been due to poor quality reference signal which would have background noise added to the primary noise because of its large distance from the noise source. Other cause could be the bandwidth of the broadband noise being too wide for the system to be able to control it.

Chapter 6

Conclusions

In this thesis it was aimed to implement an ANC headset system with a DSP developer board, the C6713 DSK housing a floating-point DSP TMS320C6713. It was also expected to develop an analogue circuit to amplify the input/output signals to/from the C6713 board. This system would then be used to experiment with various system configurations and algorithms, and with their parameters in order to achieve good attenuation, stability and robustness.

First, the analogue system was designed and produced and the software for the DSP was developed. During this process knowledge of electronic systems were reinforced with these practical aspects of the field. The circuits performed as expected, however a smaller amount electrical noise would have been better. With the software development, a new perspective about embedded systems and firmware development was obtained. The algorithms did not display any software problems but a optimal implementation, for instance of the buffers, would improve the quality of the system.

The FxLMS was the first algorithm to be developed. The active headset system, with both feedback and feedforward configurations, showed satisfactory attenuation. Not only it reinforced the fact that feedforward systems perform better than feedback systems, but also demonstrated the superior attenuation of an adaptive controller compared to some fixed controllers, seen in the consulted literature. However it revealed instability issues related to two reasons, the modelling error of the secondary path estimates, inherent to off-line modelling method, and the high sampling rate used.

To mitigate the effects of the first, the properties of both high and low sampling rates were exploited with the multi-rate processing technique. Such technique confirmed the importance of sampling rate and bandwidth on the stability and attenuation of an ANC system.

To reduce the effects of the phase error two approaches were used, one for the feedback system and the other for the feedforward system. The first was turning the system off for the frequencies where the error was more frequent and the other on-line modelling the secondary path transfer function. Thus, two of the used algorithms are derived, the BMFxLMS and the MMFxLMS. In terms of stability the algorithms met their objective, but at the cost of reduced attenuation for real noises, and other adverse effects such as the overshoot observed in the MMFxLMS. The employment of these algorithms was interesting for another reason, they are state of the art algorithms developed at INESC-ID and real time experiments

with them should be useful to further their investigation.

These algorithms were further studied by varying some of their parameters and noting the performance. This was a method of optimization of the system to achieve better results in the cancellation of real signals. However, it is incomplete procedure, therefore the parameters used were not optimal. The optimization process described should be seen as a jumping off point to further improve the quality of the system. For instance, as seen during the optimization of the feedforward FxLMS, with a bigger number of combinations of parameters tested, an unstable step size could be stabilized by another parameter. Also, it would be interesting to test all the step sizes with louder noises, to improve the stability of the system and reach for higher attenuations.

During the optimization procedure, some concepts of LMS algorithms were demonstrated. They were the relation of convergence speed and step size, the stability bound of NLMS algorithms, the property of the leakage factor of limiting the output power and of improving stability under high sound pressure noises and the convergence time of NLMS with the filter order ($n_c = 2.3L/\mu$).

The results obtained have indicated that an adaptive ANC headset system can be considered for commercial purposes. However, tracking and stability performances have to be improved before reaching the end-user. In this study, a basis for a prototype was assembled but it needs additional steps and more research to be ready to be sold in the market.

6.1 Future Work

To obtain a better active headset system prototype, closer to being a consumer product, there could be several improvements made to the system. First is to integrate all the hardware in one PCB, with low power consumption and low electrical noise, and develop an optimized firmware (for instance with circular buffering instead of a standard buffer). Another good feature would be including methods to turn the system off and on when instability is detected. To improve the control of broadband noises and transient signals an hybrid system with fixed and adaptive control could be considered. Thus the disadvantage of the slowness of adaptive controllers could be compensated. Then, some additional features, such as audio integration or communication, could be added to the system.

The study of ANC algorithms could be furthered by implementing state of the art IIR algorithms using the least square approach, other on-line secondary path modelling algorithms. Also, using a head and torso simulator to perform the system optimization and performance assessment in more diverse situations and with more combinations of parameters.

Bibliography

[1] D. M. S.M. Kuo. Active noise control: a tutorial review. *Proceedings of the IEEE*, 87(6):943–973, June 1999.

[2] M. M. S.A. Stansfeld. Noise pollution: non-auditory effects on health. *Br Med Bull*, 68(1):243–257, Dec. 2003.

[3] D. M. S.M. Kuo. *Active Noise Control Systems: Algorithms and DSP Implementations*. New York: Wiley, 1^{st} edition, 1996.

[4] e. a. S.M. Kuo, I. Panahi. Design of active noise control systems with the tms320 family. Technical report, Texas instruments, June 1996.

[5] C. Harris. *Handbook of Acoustical Measurements and Noise Control*. McGraw-Hill, 3^{rd} edition, 1991.

[6] I. V. L.L. Beranek. *Noise and Vibration Control Engineering: Principles and Applications*. New York: Wiley, 2^{nd} edition, 2005.

[7] P. N. S.J. Elliot. Active noise control. *IEEE Signal Processing Magazine*, 10(4):12–35, Oct. 1993.

[8] S. E. P.A. Nelson. *Active Control of Sound*. Academic Press, 1^{st} edition, 1992.

[9] S. Elliot. Down with noise [active noise control]. *IEEE Spectrum*, 36(6):54–61, June 1999.

[10] P. Lueg. Process of silencing sound oscillations, June 1936. US Patent 2,043,416.

[11] E. M. H.F Olson. Electronic sound absorver. *J. Acoust. Soc. Am.*, 25:1130–1136, Oct. 1953.

[12] C. Hansen. *Active Control of Noise and Vibration*. CRC Press, 1^{st} edition, 2012.

[13] T. Bourk. Noise cancellation headset, Jan. 1993. US Patent 5,182,774.

[14] H. Gether. Designing a feedback anc headset using as3435. Technical report, ams AG, 2015.

[15] J. Gehlin. Practical considerations for wired headset designs. Technical report, ams AG, 2014.

[16] J. Burgess. Active adaptive sound control in a duct: A computer simulation. *J. Acoust. Soc. Am.*, 70(3):715–726, Sept. 1981.

[17] S. S. B. Widrow. *Adaptive Signal Processing*. Prentice-Hall, 1^{st} edition, 1985.

[18] L. Eriksson. Active sound and vibration control using adaptive digital signal processing. In *Acoustics, Speech, and Signal Processing, ICASSP-93.*, volume 1, pages 51–54, April 1993.

[19] *AS3501 AS3502: Low Power Ambient Noise-Cancelling Speaker Driver*. ams AG, 2009.

[20] M. T. Akhtar and W. Mitsuhashi. Improving performance of hybrid active noise control systems for uncorrelated narrowband disturbances. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2058–2066, Sept 2011.

[21] Y. Song, Y. Gong, and S. M. Kuo. A robust hybrid feedback active noise cancellation headset. *IEEE Transactions on Speech and Audio Processing*, 13(4):607–617, July 2005.

[22] T. Schumacher, H. Krüger, M. Jeub, P. Vary, and C. Beaugeant. Active noise control in headsets: A new approach for broadband feedback anc. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 417–420, May 2011.

[23] P. Lopes. Active noise control algorithms with reduced channel count and their stability analysis. *Signal Processing*, 88(4):811–821, Apr. 2008.

[24] M. P. P.A.C. Lopes, J. Gerald. The mmfxlms algorithm for active noise control with on-line secondary path modelling. *Digital Signal Processing*, 60:75–80, Jan. 2017.

[25] A. Zaknich. *Principles of Adaptive Filters and Self-learning Systems*, chapter 1 - Adaptive Filtering. Springer, 2005.

[26] S. Douglas. *Digital Signal Processing Handbook*, chapter 18 - Introduction to Adaptive Filters. CRC Press, 2009.

[27] M. M. B. Kovačević, Z. Banjac. *Adaptive Digital Filters*, chapter 2 - Adaptive Filtering. Springer, 2012.

[28] S. Haykin. *Adaptive Filter Theory*. Prentice-Hall, 3^{rd} edition, 1996.

[29] B. W. M. Kamenetsky. A variable leaky lms adaptive algorithm. In *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004*, volume 1, pages 125–128, November 2004.

[30] D. Morgan. An analysis of multiple correlation cancellation loops with a filter in the auxiliary path. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):454–467, Aug 1980.

[31] S. Elliot. *Signal Processing for Active Control*, chapter 3.4.5 - Adaptive Filtering. Academic Press, 2000.

[32] G. C. Carter. Coherence and time delay estimation. *Proceedings of the IEEE*, 75(2):236–255, Feb 1987.

[33] J. L. M.R. Bai, Y. Lin. Reduction of electronic delay in active noise control systems— a multirate signal processing approach. *The Journal of the Acoustical Society of America*, 111(2):916–924, Nov 2002.

[34] X. Q. L. Zhang. Causality study on a feedforward active noise control headset with different noise coming directions in free field. *Applied Acoustics*, 80:36–44, Jun 2014.

[35] C. Hansen. *Understanding Active Noise Cancellation*. CRC Press, 1^{st} edition, 2001.

[36] X. Yu, S. Gujjula, and S. M. Kuo. Active noise control for infant incubators. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2531–2534, Sept 2009.

[37] P. Thanigai, S. M. Kuo, and R. Yenduri. Nonlinear active noise control for infant incubators in neonatal intensive care units. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 1, pages I–109–I–112, April 2007.

[38] L. Liu, S. M. Kuo, and M. Zhou. Virtual sensing techniques and their applications. In *Networking, Sensing and Control, 2009. ICNSC '09. International Conference on*, pages 31–36, March 2009.

[39] B. C. D.P. Das, D.J. Moreau. A computationally efficient frequency-domain filtered-x lms algorithm for virtual microphone. *Mechanical Systems and Signal Processing*, 37(1-2):440–454, June 2013.

[40] M. Bouchard and S. Quednau. Multichannel recursive-least-square algorithms and fast-transversal-filter algorithms for active noise control and sound reproduction systems. *IEEE Transactions on Speech and Audio Processing*, 8(5):606–618, Sep 2000.

[41] C.-Y. Chang and F.-B. Luoh. Enhancement of active noise control using neural-based filtered-x algorithm. *Journal of Sound and Vibration*, 305(1–2):348 – 356, Aug 2007.

[42] S. Ahmed, M. T. Akhtar, and X. Zhang. Robust auxiliary-noise-power scheduling in active noise control systems with online secondary path modeling. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(4):749–761, April 2013.

[43] A. Carini and S. Malatini. Optimal variable step-size nlms algorithms with auxiliary noise power scheduling for feedforward active noise control. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(8):1383–1395, Nov 2008.

[44] K. Fujii and J. Ohga. Method to update the coefficients of the secondary path filter under active noise control. *Signal Processing*, 81(2):381 – 387, 2001.

[45] K. Fujii, M. Muneyasu, and J. Ohga. Active noise control system using the simultaneous equation method without the estimation of error path filter coefficients. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 85(12), 2002.

[46] T. Brittain. Active noise reduction headset, Oct. 1997. US Patent 5,675,658.

[47] M. Hsieh. Active noise cancellation earphone, Dec. 2010. US Patent App. 12/816,806.

[48] M. Bergeron, S. Crump, and D. Gauger. Noise reduction headset, May 2012. US Patent 8,189,803.

[49] W. S. Gan and S. M. Kuo. Integrated active noise control communication headsets. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 4, May 2003.

[50] W. S. Gan and S. M. Kuo. An integrated audio and active noise control headset. *IEEE Transactions on Consumer Electronics*, 48(2):242–247, May 2002.

[51] W. G. S.M.Kuo, S. Mitra. Active noise control system for headphone applications. *IEEE Transactions on Control Systems Technology*, 14(2):331–335, March 2006.

[52] H. S. Vu and K. H. Chen. A low-power broad-bandwidth noise cancellation vlsi circuit design for in-ear headphones. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 24(6): 2013–2025, June 2016.

[53] L. Liu, S. M. Kuo, and K. P. Raghuathan. An audio integrated motorcycle helmet. *Journal of Low Frequency Noise, Vibration and Active Control*, 29(3):161–170, Sept. 2010.

[54] C. Y. Chang and S. T. Li. Active noise control in headsets by using a low-cost microcontroller. *IEEE Transactions on Industrial Electronics*, 58(5):1936–1942, May 2011.

[55] B. Rudzyn and M. Fisher. Performance of personal active noise reduction devices. *Applied Acoustics*, 73(11):1159 – 1167, 2012.

[56] *TMS320C6713: Floating-Point Digital Signal Processor*. Texas Instruments, 2005.

[57] *TLV320AIC23B: Stereo Audio CODEC, 8- to 96-kHz, with integrated headphone amplifier*. Texas Instruments, 2004.

[58] *16-Bit, Low-Power Stereo Audio CODEC With Microphone Bias, Headphone, and Digital Speaker Amplifier*. Texas Instruments, 2007.

[59] *TMS320C6713 DSK Technical Reference*. Spectrum Digital Inc., 2003.

[60] T. Engdahl. Speaker impedance, 1999. URL `http://www.epanorama.net/documents/audio/speaker_impedance.html`. Accessed: 2016-04-26.

[61] How to bias an op-amp, 2011. MIT OpenCourseWare.

[62] *Electret Condenser Microphone*. Pro Signal, 2015.

[63] *Omni-Directional Foil Electret Condenser Microphone*. Challenge Electronics, 2010.

[64] J. Eargle. *The Microphone Book*, chapter 7 - Microphone Measurements, Standards, and Specifications. Focal Press, 2004.

[65] *Sound system equipment - Part 4: Microphones*. International Electrotechnical Comission, 2014. IEC 60268-4:2014.

[66] *MCP6286 : Low Noise, Low Power Op Amp*. Microchip, 2009.

[67] *TS4871: Output rail to rail 1W audio power amplifier with standby mode*. STMicroelectronics, 2003.

[68] S. Incorporated. Understanding earphone / headphone specifications. URL http://shure.custhelp.com/app/answers/detail/a_id/2991/~/understanding-earphone-%2F-headphone-specifications. Accessed: 2016-05-11.

[69] S. Keeping. Understanding the advantages and disadvantages of linear regulators, 2012. URL http://www.digikey.pt/en/articles/techzone/2012/may/understanding-the-advantages-and-disadvantages-of-linear-regulators. Accessed: 2016-09-10.

[70] *uA78xx Fixed Positive Voltage Regulators*. Texas Instruments, 2015.

[71] Hugo. Tips on how to properly design/layout a printed circuit board (pcb). URL http://www.onmyphd.com. Accessed: 2016-04-20.

[72] P. Loughhead. Via stitching, 2016. URL http://techdocs.altium.com/display/ADOH/Via+Stitching. Accessed: 2016-07-16.

[73] *X-NM1 Operating Instructions*. Pioneer Corporation, 2001.

[74] 1 hour of industrial air compressor sounds. URL https://www.youtube.com/watch?v=Xj5X-ydtyI4. Accessed: 2016-09-26.

[75] Brown noise. URL https://soundcloud.com/onlinetonegenerator/brown-noise. Accessed: 2016-09-26.

[76] A. Tokatli. Design and implementation of a dsp based active noise controler for headsets. Master's thesis, The Middle East Technical University, August 2004.

[77] W. R. A.K. Wang, B. Tse. Adaptive active noise control for headphones using the tms320c30 dsp. Technical report, Texas instruments, January 1997. spra160.

Appendix A

Multi-rate signal processing

Multi-rate digital signal processing consists of decimation, interpolation and resampling (which is the combination of the two) processes.

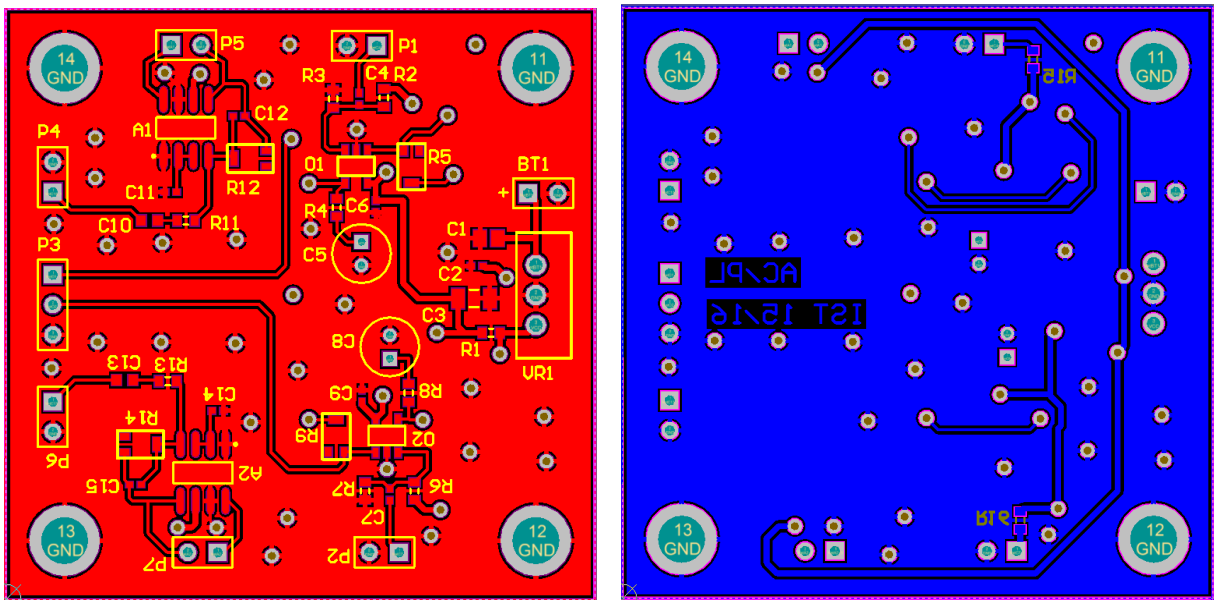
Decimation is the decrease of sampling frequency. It implies low-pass filtering a signal and then throwing away some of its samples (downsampling). The filtering is done to eliminate signal components with frequency higher than half of the post-decimation sample rate, $f > fs_{2M}$, in order to avoid aliasing. This may lead to information loss if there are frequency components higher than the Nyquist rate. To implement the downsampling the Mth sample is kept and the M-1 samples in between are thrown away.

Interpolation is the increase of sampling rate and it is done by upsampling followed by filtering. Upsampling occurs by inserting L-1 zero-valued samples between the original samples to increase the sampling rate. This process adds to the original signal undesired spectral images which are centred on multiples of its sampling rate. To keep these images out of the post-upsampling Nyquist band, which is the interval [0, $\frac{Lfs}{2}$] Hz, it is necessary to use an anti-image digital filter to eliminate these signal components with frequency in the interval [$\frac{fs}{2}$,$\frac{Lfs}{2}$]Hz.

Interpolation leads to energy loss of the signal by a factor of L. This happens because the upsampling compresses the signal spectrum, since its energy is spread with the repetitions. After their removal the spectrum of the post interpolation signal is equal to the original signal however compressed. If one desires to maintain the signal energy level throughout the system, a gain equal to the conversion factor must be used at the output.

Appendix B

Analogue circuit PCB Design



(a) Top layer

(b) Bottom layer

Figure B.1: PCB layout.

Table B.1: Bill of Materials for PCB.

Qty	Value	Designator	Description	Manufacturer	Part Number
1	0.33 μF	C1	SMD Multilayer Ceramic Capacitor, MC Series, 0.33 μF, $\pm$ 10%, X7R, 16 V, 0805	Multicomp	MC0805B334K160CT
3	0.1 μF	C2, C6, C9	SMD Multilayer Ceramic Capacitor, MC Series, 0.1 μF, $\pm$ 20%, X5R, 10 V, 0402	Multicomp	MC0402X104M100CT
4	1 μF	C4, C7, C11, C14	SMD Multilayer Ceramic Capacitor, GRM Series, 1 μF, $\pm$ 10%, X6S, 6.3 V, 0402	Murata	GRM155C80J105KE15D
2	10 μF	C5, C8	Aluminium Electrolytic Capacitors - Leaded 50V 10uF 20% 5x11mm	Lelon	REA100M1HBK-0511P
2	82 nF	C10, C13	SMD Multilayer Ceramic Capacitor, CC Series, 0.082 μF, $\pm$ 10%, X7R, 16 V, 0603	Yageo	CC0603KRX7R7BB823
2	47 pF	C12, C15	SMD Multilayer Ceramic Capacitor, MC Series, 47 pF, $\pm$ 5%, C0G / NP0, 50 V, 0402	Multicomp	MC0402N470J500CT
6	100kΩ	R2, R3, R6, R7, R11, R13	Surface Mount Chip Resistor, Thick Film, MCWR Series, 100 kohm, 100 mW, $\pm$ 1%, 50 V	Multicomp	MCWR06X1003FTL
2	3.6kΩ	R4, R8	Surface Mount Chip Resistor, Thick Film, MCWR Series, 3.6 kohm, 100 mW, $\pm$ 1%, 50 V	Multicomp	MCWR06X3601FTL
2	5.1kΩ	R15, R16	Surface Mount Chip Resistor, Thick Film, MCWR Series, 5.1 kohm, 100 mW, $\pm$ 1%, 50 V	Multicomp	MCWR06X5101FTL
4	200kΩ	R5, R9, R12, R14	TRIMMER, 200KOHM, 25%, 0.15W	Panasonic Electronic Components	EVM2WSX80B25
2		O1, O2	Operational Amplifier, Single, AEC-Q100, 3.5 MHz, 1 Amplifier, 2 V/μs, 2.2V to 5.5V, SOT-23, 5 Pins	Microchip	MCP6286T-E/OT
2		A1, A2	Audio Power Amplifier, AB, 1 Channel, 1 W, 2.5V to 5.5V, SOIC, 8 Pins	STMicroelectronics	TS4871ID
1		VR1	Linear Voltage Regulator, Fixed, 7V To 25V In, 5V And 1.5A Out, TO-220-3	Texas Instruments	UA7805CKCT

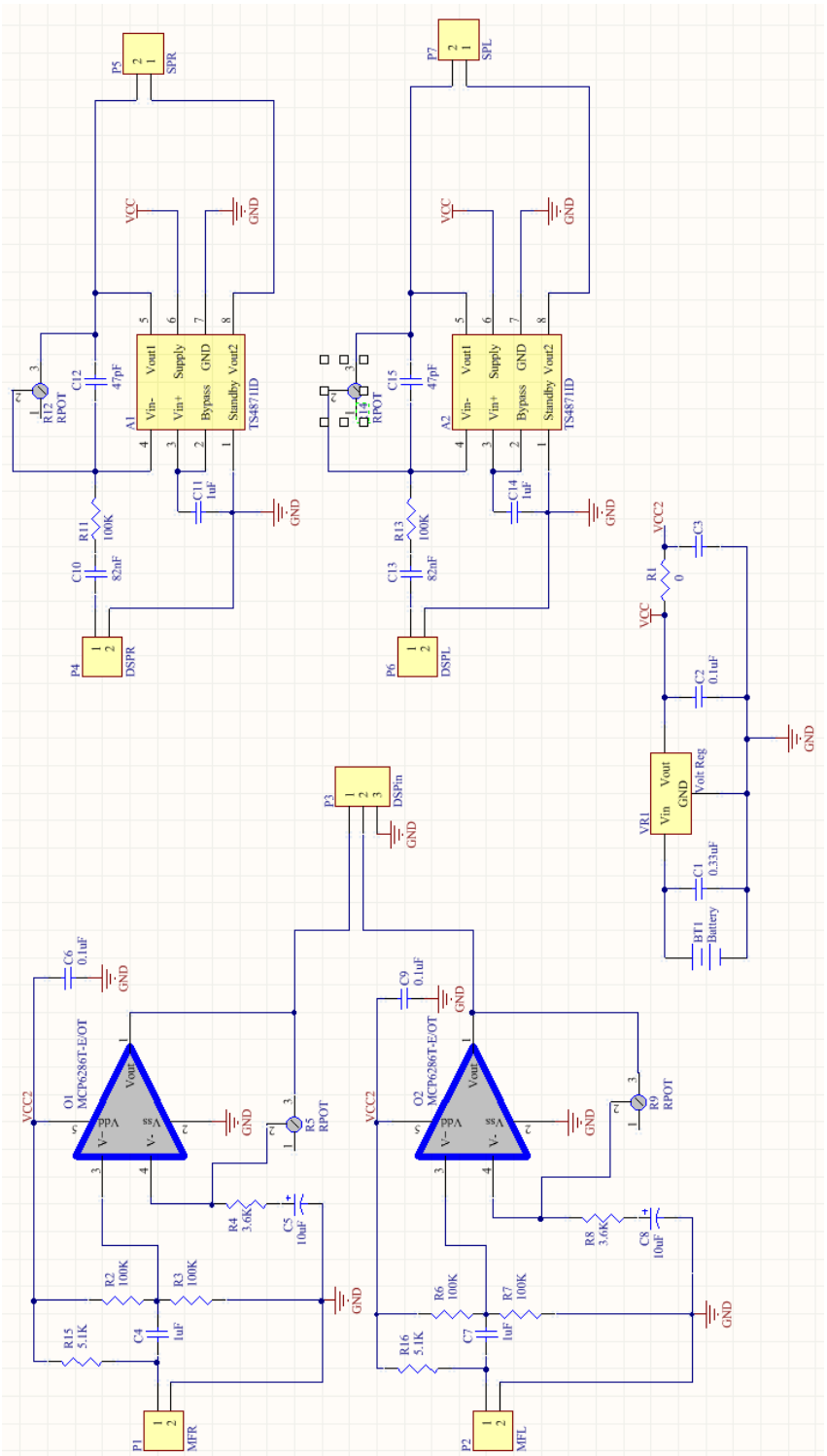


Figure B.2: PCB schematic

Appendix C

Primary noise variation results

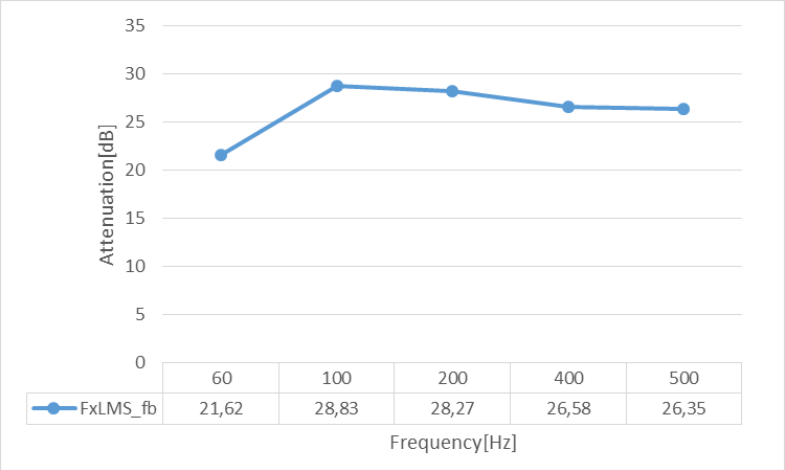


Figure C.1: Attenuation of FxLMS as a function of the frequency of a tone.

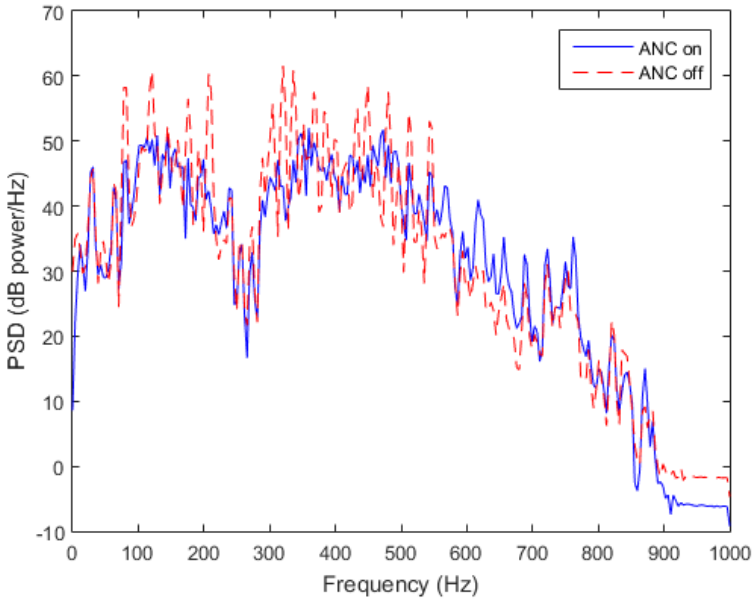


Figure C.2: Power spectra of narrowband noise with and without ANC, for FxLMS.

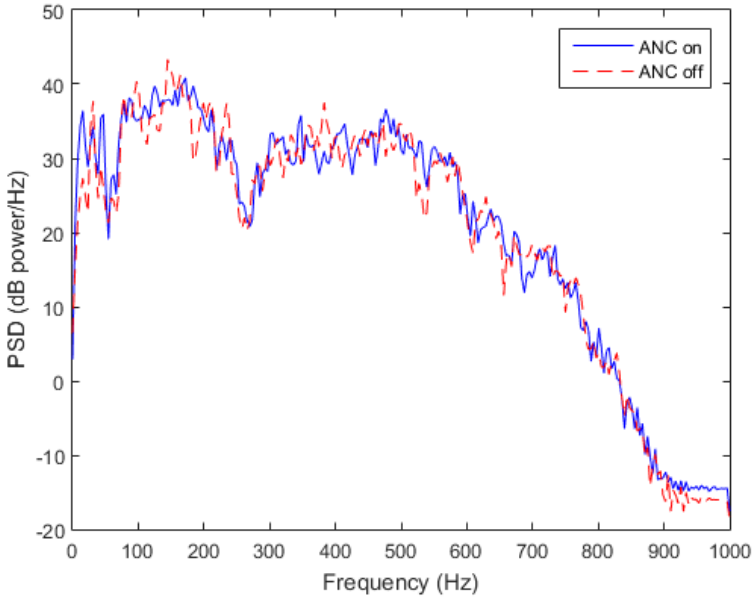


Figure C.3: Power spectra of broadband noise with and without ANC, for FxLMS.

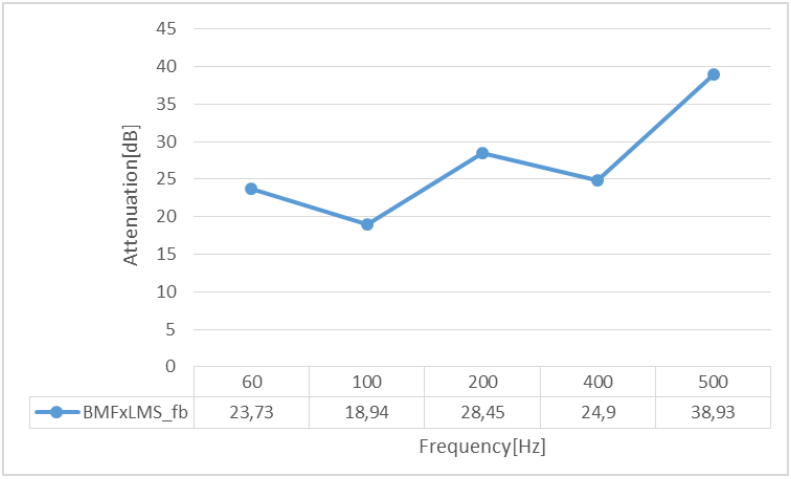


Figure C.4: Attenuation of BMFxLMS as a function of the frequency of a tone.

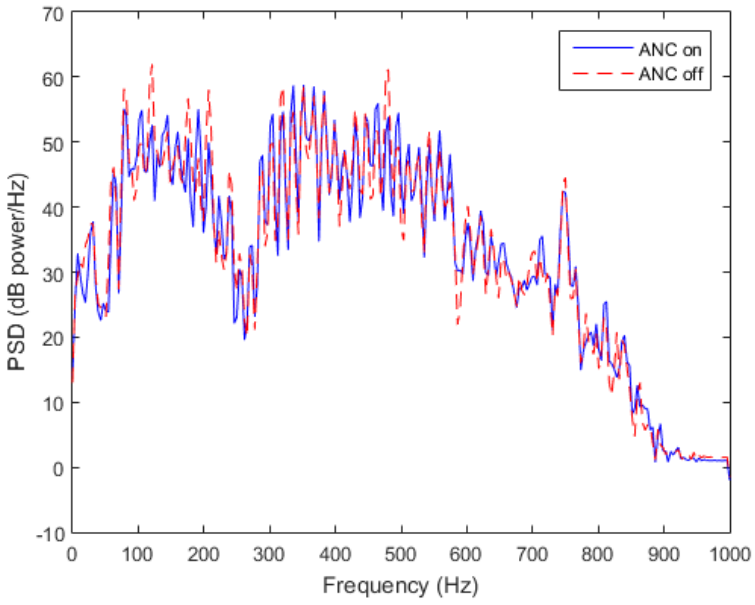


Figure C.5: Power spectra of narrowband noise with and without ANC, for BMFxLMS.

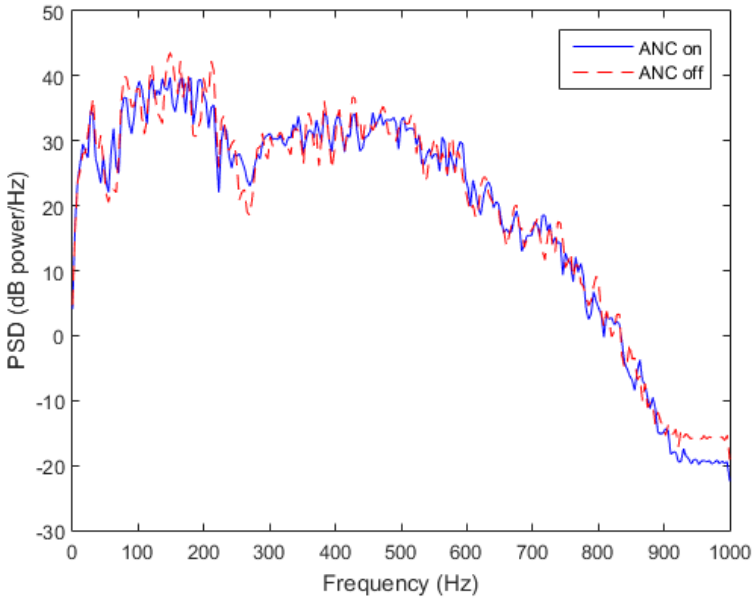


Figure C.6: Power spectra of broadband noise with and without ANC, for BMFxLMS.

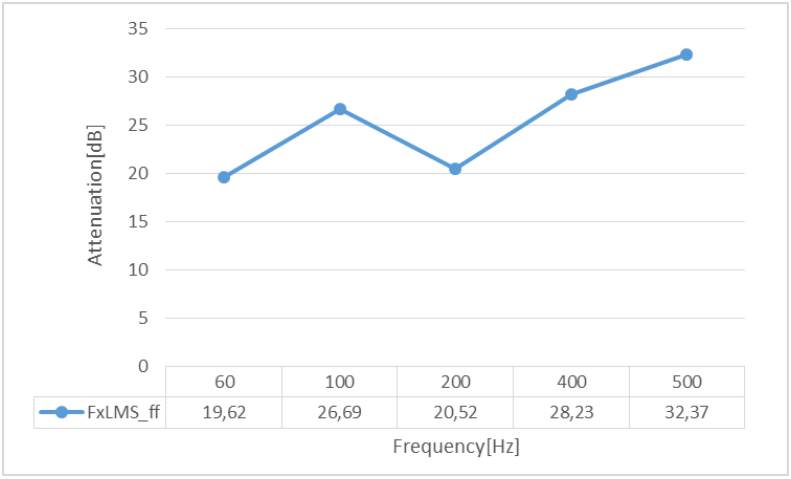


Figure C.7: Attenuation of feedforward FxLMS as a function of the frequency of a tone.

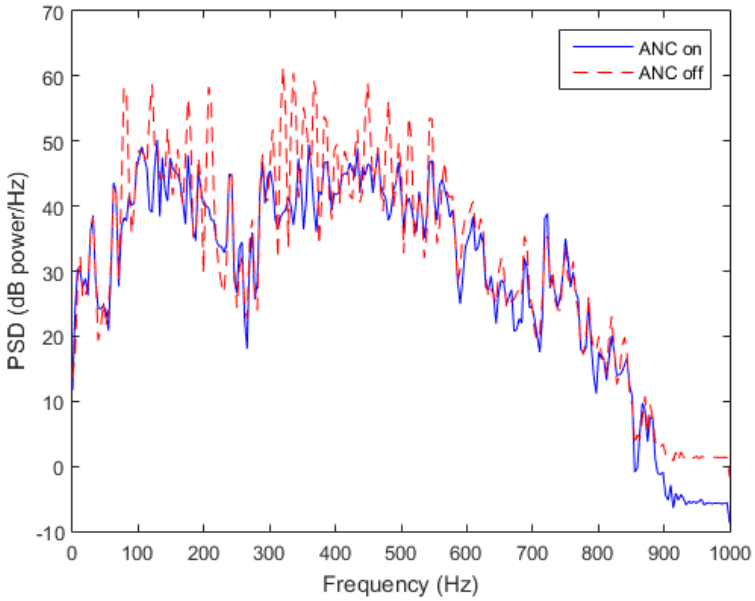


Figure C.8: Power spectra of narrowband noise with and without ANC, for feedforward FxLMS.

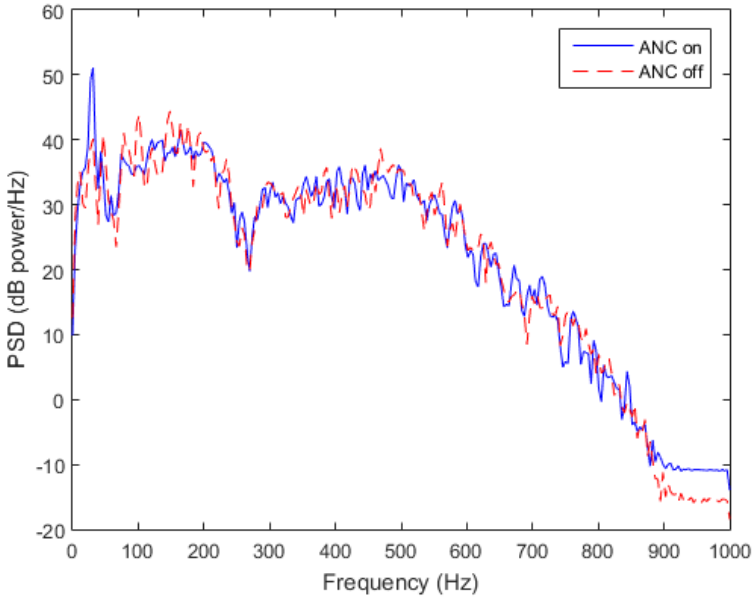


Figure C.9: Power spectra of broadband noise with and without ANC, for feedforward FxLMS.

Table C.1: Attenuation and Overshoot of the frequency sweep for MMFxLMS.

Frequency[Hz]	Att[dB]	Ovs[dB]
60	16,62	17,95
80	27,53	1,76
100	29,6	1,582
200	29,72	22,11
400	23,77	6,02
500	25,03	15,89

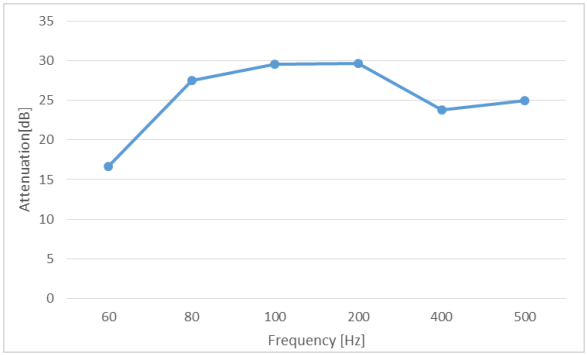


Figure C.10: Attenuation of MMFxLMS as a function of the frequency of a tone.

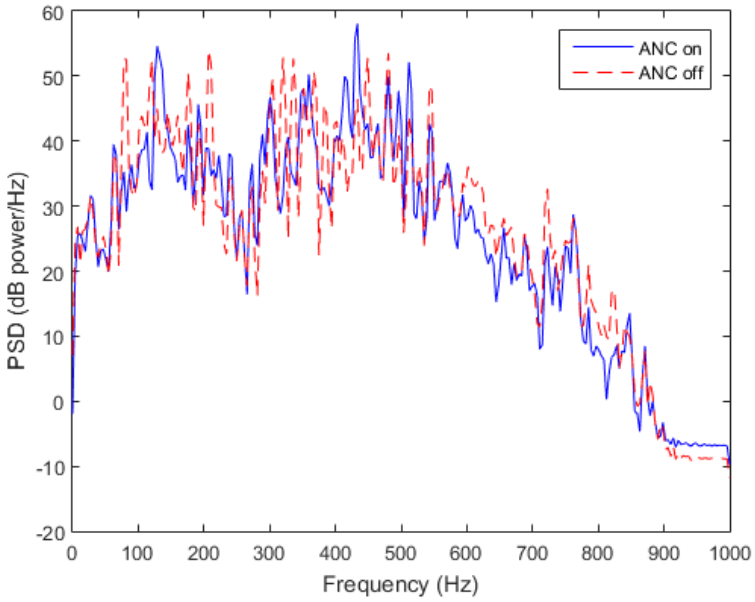


Figure C.11: Power spectra of narrowband noise with and without ANC, for MMFxLMS.

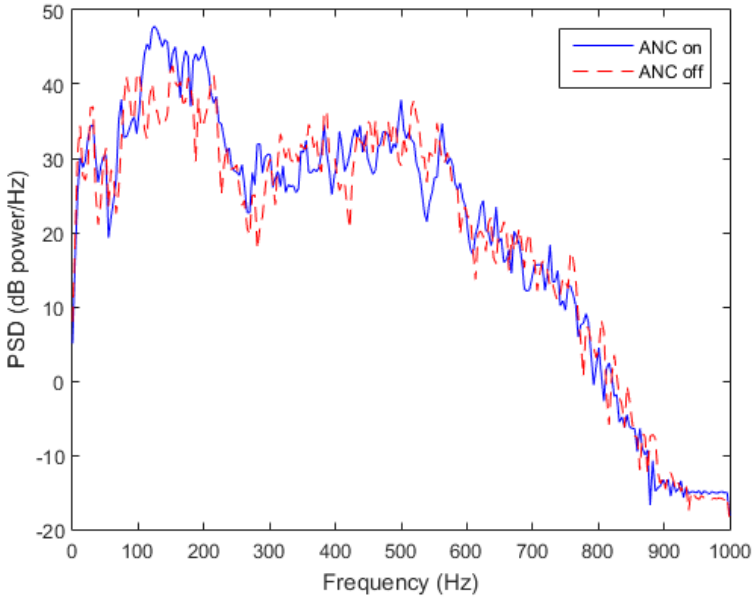


Figure C.12: Power spectra of broadband noise with and without ANC, for MMFxLMS.