Active Noise Control in Forest Machines

Fredrik Forsgren

September 2, 2011
Master’s Thesis in Engineering Physics, 30 credits
Supervisor: Magnus Berggren and Fredric Lindström
Examiner: Erik Fällman

UMEÅ UNIVERSITY
DEPARTMENT OF PHYSICS
SE-901 87 UMEÅ
SWEDEN
Abstract

Achieving a low noise level is of great interest to the forest machine industry. Traditionally this is obtained by using passive noise reduction, i.e. by using materials for sound isolation and sound absorption. Especially designs to attenuate low frequency noise tend to be bulky and impractical from an installation point of view. An alternative solution to the problem is to use active noise control (ANC). The basic principle of ANC is to generate an anti-noise signal designed to destructively interfere with the unwanted noise.

In this thesis two algorithms (Feedback FxLMS and Feedforward FxLMS) are implemented and evaluated for use in the ANC-system. The ANC-system is tuned to the specific environment in the driver’s cabin of a Komatsu forest machine. The algorithms are first tested in a simulated environment and then in real-time inside a forest machine.

Simulations are made both in Matlab and in C using both generated signals and recorded signals. The C code is implemented on the Analog Devices Blackfin DSP card BF526.

The result showed a significantly reduction of the sound pressure level (SPL) in the driver’s cabin. The noise attenuation obtained using the Feedback FxLMS was approximately 14 dB for a tonal 100 Hz signal and 11 dB using recorded engine noise from a forest machine at 850 rpm.

Keywords: Active Noise Control (ANC), Forest machine, Adaptive signal processing, Noise reduction, Adaptive filter, Feedforward, Feedback, Leaky, NLMS, FxLMS
## Contents

Abbreviations and Symbols ix

1 Introduction 1
   1.1 Problem Statement and Objectives 2
      1.1.1 Goals 3
   1.2 Related Work 3

2 Theory 5
   2.1 Basic principle of ANC 5
   2.2 FIR filter and Notation 6
   2.3 The Steepest Descent Adaptive Filter 8
   2.4 Least Mean Square 10
   2.5 Normalized LMS 11
   2.6 Leaky LMS 12
   2.7 Filtered-x LMS Algorithm 12
   2.8 Fixed-Point Arithmetic 13
   2.9 Real-Time Constraints 14

3 Methods 17
   3.1 Secondary Path Identification 17
      3.1.1 Algorithm 18
   3.2 Feedforward ANC 19
      3.2.1 Algorithm 20
   3.3 Feedback ANC 20
      3.3.1 Algorithm 20

4 Results 23
   4.1 Secondary Path 23
   4.2 ANC performance with tonal noise 25
   4.3 ANC performance with engine noise 27
   4.4 SPL measurements 29
5 Conclusions and Future Work 31

6 Acknowledgments 33

References 35

A Blackfin BF526 37
A.1 Tables 38
### List of Figures

2.1 Principle of sound cancellation - Interference of noise and anti-noise ........... 5  
2.2 Block diagram of FIR filters in time-domain and z-domain. ......................... 8  
2.3 Block diagram of an adaptive filter. .............................................. 8  
2.4 Example of the Mean Square Error $\xi(n) = E\{e(n)^2\}$ of two filter weights is illustrated by a surface plot to the left and a contour plot to the right. The direction of the steepest descent points towards the minimum of the error surface (opposite to the gradient). ..................................................... 10  
2.5 Block diagram of Filtered-x LMS ................................................. 13  
3.1 Block diagram of Secondary Path Identification ................................... 18  
3.2 Block diagram showing the principle of Feedforward sound cancellation ... 19  
3.3 Block diagram showing the principle of Feedback sound cancellation ....... 21  
4.1 Time plot of the Secondary path. .................................................. 24  
4.2 Time plot of the error signal with a tonal noise of 100 Hz. The plot to the right is with ANC and to the left is without ANC. ................................. 25  
4.3 Power Spectral Density Estimate of the error signal with a tonal noise of 100 Hz. The solid line is when the ANC-system is turned on and dashed line when turned off. ............................................................... 26  
4.4 Time plot of the error signal at the engine speed 850 rpm. The plot to the right is with ANC and to the left is without ANC. ................................. 27  
4.5 Power Spectral Density Estimate of the error signal at the engine speed 850 rpm. The solid line is when the ANC-system is turned on and dashed line when it is turned off. ............................................................... 28  
A.1 Blackfin BF526 ................................................................................. 37
List of Tables

4.1 Sound pressure levels measured with the decibel meter . . . . . . . . . . . . . 29
4.2 Sound pressure levels measured with the error sensor . . . . . . . . . . . . . 29

A.1 Fostex PM0.4n . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
A.2 Blaupunkt GTb200A . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
A.3 Blaupunkt THb200A . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 39
A.4 ECM-304BD . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 39
Abbreviations and Symbols

$T$  
Sampling time

$X(n)$  
z-transform of $x(n)$

$\omega$  
Angular frequency

$x(n)$  
$[x(n), x(n-1), \ldots, x(n-p)]^T$ vector of past samples of signal $x(n)$

$f_s$  
Sample rate

$x(n)$  
Sampled scalar time signal as a function of the discrete time variable $n$

$\gamma$  
Leaky factor

$Z$  
z-transform

$d$  
Desired signal

$e$  
Error signal

$*$  
Convolution

$\beta$  
Normalized step size used in NLMS

$\in$  
Set membership

$Z$  
Integers

$Z$  
z-transform

$Z^{-1}$  
Inverse z-transform

$\mu$  
Step size used in LMS

$\nabla$  
The gradient. In this case the partial derivatives with respect to the filter weights.

$\| \|_{\ell_2}$  
$L^2$ norm, e.g. $\|x\|_{\ell_2} = \sqrt{x^2(n) + x^2(n-1) + \ldots + x^2(n-p)}$

$\omega$  
Angle of $z$
\( h \)  
\[ [h(0), h(1), \ldots, h(N - 1)]^T \] vector of the filter weights.

\( w_n \)  
\[ [w_n(0), w_n(1), \ldots, w_n(p)]^T \] vector of the adaptive filter weights.

\( \xi \)  
Mean-square error function

\( c_0 \)  
Speed of sound

\( d \)  
Desired signal

\( e \)  
Error signal

\( E_x \)  
Energy of the input signal \( x(n) \)

\( E \)  
Expected value

\( f_M \)  
Maximum sampling frequency

\( f_s \)  
Sampling frequency

\( h \)  
Impulse response

\( H(z) \)  
Impulse response in the z-domain

\( j \)  
Imaginary unit

\( L \)  
Distance between loudspeaker and microphone (Feedback LMS)

\( n \)  
Discrete time variable, \( n \in \mathbb{Z} \)

\( p \)  
FIR filter order

\( t_p \)  
Processing time

\( w_n \)  
Weight of the adaptive filter

\( x \)  
Input signal

\( X(z) \)  
Input signal in the z-domain

\( y \)  
Output signal

\( Y(z) \)  
Output signal in the z-domain

\( z \)  
A complex variable, \( z = re^{j\omega} \)

ANC  
Active Noise Control

Codec  
Coding and decoding

DSP  
Digital Signal Processor

FIR  
Finite Impulse Response
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FxLMS</td>
<td>Filtered X Least Mean Square</td>
</tr>
<tr>
<td>LMS</td>
<td>Least Mean Square</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Square Error</td>
</tr>
<tr>
<td>NLMS</td>
<td>Normalized Least Mean Square</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

In todays forest machine industry it is important to create a comfortable noise level for the driver. Traditionally this is obtained by using passive noise reduction, i.e. by using materials for sound isolation and sound absorption. To obtain a good result with passive noise reduction, the solution tend to be bulky and expensive. Especially designs to attenuate low frequency noise can be impractical from an installation point of view. For example the absorption coefficient of 50 mm thick fiber-glass is 0.94 at 4000 Hz and only 0.17 at 125 Hz [1]. Therefore, alternative solutions of the low frequency problem are of great interest. A well known alternative solution is to use active noise control (ANC). The basic principle of ANC is to use an sound source to generate an anti-noise signal designed to destructively interfere with the unwanted noise. An ANC-system performs best at low frequencies. Low frequencies enables the use of lower sampling frequency which result in a lower computational load. Also a larger region around the driver’s head can be controlled by the ANC-system, since the noise cancellation region depends strongly on the frequency [5]. A combination of passive and active noise reduction is therefore a good choice. The noise levels in forest machines affects the driver in several aspects. Noise can be hurtful, annoying, decrease the attention of other signals, cause sleep disorder and decrease the job performance etc. Research has been made to analyze whether ANC influences the driver’s physiological and psychological responses. The study "Effects of Active Noise Control on Physiological Functions" [9] verified that ANC reduces the stress on the human body, especially the cardiovascular system. Significant changes in systolic blood pressure, diastolic blood pressure, sympathetic nerve activity, and parasympathetic nerve activity was measured.
Already in the 15\textsuperscript{th} century Leonardo Da Vinci drawing sketches illustrating the destructive force of canceling waveforms. He studied how water in two canals can cancel each other when they meet if the timing (phase) is correct \cite{12}. The first patent of ANC was granted in 1936. It was when Paul Lueg proposed a design of an acoustic ANC-system for controlling sound in a duct. His system contained a microphone to detect the noise, an electronic controller to adjust the signal, and a loudspeaker to produce the canceling signal in the duct \cite{10}. In 1953 Olson and May presented the first demonstration of ANC by creating a quiet zone in the area surrounding an individual’s head \cite{11}. These solutions works in a non-stationary surrounding, but to adjust such time dependent variation an adaptive ANC-system is needed. Later in 1960 the least mean-square (LMS) algorithm was introduced which could solve this problem. The algorithm was invented at Stanford University by professor Bernard Widrow and his first Ph.D. student, Ted Hoff \cite{15}. The next big step was the filtered-x LMS (FxLMS) algorithm which was originally proposed by Morgan 1980 and independently for feedforward control by Widrow 1981 and for the active control of sound in ducts by Burgess 1981 \cite{2}. The FxLMS algorithm compensate for the physical path between the control sources and error sensors that often causes instability when using the conventional LMS algorithm. In spite of these scientific successful ideas, it would not be until the 1980’s, when the computers and microprocessors became cheaper and more powerful, that the idea would have any commercial value.

This thesis consist of 6 chapters. The first chapter gives an introduction to ANC and the problem description. The problem is stated, goals is set, purposes defined and the related work is described. The second chapter explains the theory behind ANC needed to understand the later part of the thesis such as FIR filter, transforms, LMS algorithms, fixed-point effects, and sampling etc. Third chapter describes the method used to create the prototype of the ANC-system and to test the performance. Results are presented in the the fourth chapter and in the the fifth chapter the conclusions. Finally, the sixth chapter consists of the acknowledgments.

\section{Problem Statement and Objectives}

The purpose of this thesis is to implement and evaluate algorithms for ANC and tune these to the specific environment in the driver’s cabin of a Komatsu Forest machine. The algorithms will at first be tested in a simulated environment and in real-time. The simulations will
be made both in Matlab and in C using both generated signals and recorded signals. The 
C code are going to be implemented on a digital signal processor (DSP). The real time 
testing will at first be made in a controlled environment where the noise signal is played 
on a speaker. At a later stage a real forest machine cabin will be evaluated with recorded 
engine noise. All developed algorithms will be executed in real-time on a processor with a 
limited amount of signal processing power.

1.1.1 Goals
- Implement the ANC algorithms in Matlab.
- Implement the ANC algorithms on the Analog Devices Blackfin DSP card BF526.
- Real time cancellation of tones and recorded noise in a controlled environment and in 
a Komatsu forest machine.

1.2 Related Work

Many articles have been written and even commercial products can be found on the market. 
The most successful applications of active control have been for reducing noise in enclosed 
spaces such as vehicle cabins, headphones, exhaust pipes, and ducts. Particularly when 
targeting single or multiple tonal noise [6].

The most commercial widespread product is ANC headphones. Active headphones aim 
to cancel low-frequency noise and allowing the user to hear mid- and high-frequency sounds 
such as warning sirens and conversations. Some also have the ability to allow exterior signal 
inputs such as music or voice communications. Therefore it is often used by pilots especially 
in helicopters and noisy propeller-driven aircrafts [3].

Active mufflers for industrial engine exhaust stacks has also been a commercial success. 
Several automobile manufacturers are now considering active mufflers in their cars as a 
result of the fallen prices for active automobile mufflers in recent years [8].

The application of low-frequency noise cancellation inside vehicle cabins has become pop-
ular in resent years. Most major aircraft manufacturers are now developing ANC-systems, 
especially targeting noisy propeller-driven aircrafts. One example of this is the aircraft Saab 
2000 and Saab 340B+ where ANC-system is standard equipment. This results in dramatic 
weight savings compared to only use passive noise reduction techniques [13].
Another application area is car interiors. Automobile manufacturers are considering active control for reducing low-frequency noise inside their cars. By using the built-in car stereo speakers together with an ANC-system the noise can be attenuated. Lotus is one producer that already offers these kinds of systems for sale to other automobile manufacturers [14].

Currently ANC is not widely used in forest machines. Thus during the development of the ANC-system prototype, I gathered information from work done in related areas, especially from automobile and aircraft projects. The most useful books that gave me a deeper understanding of the subject are Sen M. Kuo and Bob H. Lee’s book Real-Time Digital Signal Processing [5], Stephen Elliott’s book Signal processing for active control [2], and Statistical digital signal processing and modeling written by Monson H. Hayes [4].
Chapter 2

Theory

2.1 Basic principle of ANC

ANC is based on the superposition principle. When two waves interact, the resulting waveform depends on the frequency amplitude and relative phase of the two waves, see figure 2.1. Total cancellation will occur if the "noise" and the "anti-noise" have the same phase and opposite amplitude. The challenges are to find a fast, stable and accurate solution to identify the "noise" and generate the "anti-noise" in real time [2].

![Diagram of sound cancellation](image)

Figure 2.1: Principle of sound cancellation - Interference of noise and anti-noise
2.2 FIR filter and Notation

A finite impulse response (FIR) in an signal processing filter with a impulse response which is zero outside a finite interval. If the filter is causal, is each sample of the output sequence \( y(n) \) only dependent on the current and past values of the input sequence \( x(n), x(n-1), x(n-2), \) etc where \( n \in \mathbb{Z} \) is a time variable.

Let the filter coefficients of the impulse response be \( h(k) = 0 \) for \( k < 0 \) and \( k \geq p \) thus the output signal can be expressed as a linear combination of present and past inputs as follows

\[
y(n) = \sum_{k=0}^{p} h(k)x(n-k)
\]

where \( k \in \mathbb{Z} \) and \( p \) is the filter order [4].

The FIR filter can also be expressed as the inner product using vector notation. Thus the output signal \( y(n) \) given in Eq. 2.1 can be expressed with the vector form as

\[
y(n) = h^T x(n)
\]

where \( T \) denotes the transpose operation of the vector and the filter weight vector is

\[
h = \begin{bmatrix}
    h(0) \\
    h(1) \\
    \vdots \\
    h(N-1)
\end{bmatrix}
\]

and the input signal vector

\[
x(n) = \begin{bmatrix}
    x(n) \\
    x(n-1) \\
    \vdots \\
    x(n-p)
\end{bmatrix}
\]

A commonly used notation in signal processing is the convolution notation. The FIR filtra-
2.2. FIR filter and Notation

tion expressed in discrete linear convolution form is simply

\[ y(n) = h(n) \ast x(n) \]  \hspace{1cm} (2.3)

where \( \ast \) denotes convolution. Another way to make a FIR filtration is to transform the signal from the discrete-time domain to the z-domain. The bilateral z-transform \( X(z) \) of a discrete-time signal \( x(n) \) is defined as

\[ X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \]  \hspace{1cm} (2.4)

and the inverse z-transform is

\[ x(n) = \mathcal{Z}^{-1}\{X(z)\} = \frac{1}{2\pi} \int_{-\pi}^{+\pi} X(e^{j\omega})e^{j\omega n}d\omega \]  \hspace{1cm} (2.5)

where \( z = re^{j\omega} \) is a complex variable, \( j \) is the imaginary unit and \( \omega \) is the angle of \( z \).

A special case of the z-transform is the discrete-time Fourier transform (DTFT), which is achieved when \( z = e^{j\omega} \).

A useful property of the z-transform is the way of discrete linear convolution described by the Convolution theorem.

**Theorem 2.2.1 (Convolution theorem).** Convoluted in the z-domain becomes multiplication in the discrete-time domain.

This property makes it possible to calculate a FIR filtration in the z-domain by

\[ Y(z) = \mathcal{Z}\{y(n)\} = \mathcal{Z}\{h(n) \ast x(n)\} = H(z)X(z) \]  \hspace{1cm} (2.6)

and can be transformed to the discrete-time domain by the inverse z-transform.

Figure 2.2 illustrate a block diagram of an FIR filter both in time-domain and z-domain and also the relation between the different domains. FIR filters are always bounded-input bounded-output (BIBO) stable. If a system is BIBO stable, then the output to the system is bounded for every input that is bounded. This property makes FIR filters suitable for ANC-systems [4].
2.3 The Steepest Descent Adaptive Filter

The Steepest Descent Adaptive Filter uses a FIR filter with time varying filter weights. The objective of the algorithm is to iteratively find the optimal filter that minimizes the mean-square error (MSE) function

$$\xi(n) = E\{e(n)^2\}$$

(2.7)

where $e(n)$ is the error signal at time $n$.

![Block diagram of an adaptive filter](image)

Figure 2.3: Block diagram of an adaptive filter.

Figure 2.3 illustrate a block diagram of an adaptive filter. This adaptive filter is adjusting the weights of the filter until $y(n)$ is an satisfying estimate of the desired signal $d(n)$. The filter update is based on the input signal $x(n)$ and the error signal

$$e(n) = d(n) + y(n) = d(n) + w_n^T x(n)$$

(2.8)
where the filter weight vector is
\[ w_n = \begin{bmatrix} w_n(0) \\ w_n(1) \\ \vdots \\ w_n(p) \end{bmatrix} \]

The core of the adaptive filter is to iteratively adjusts the filter in each time step. In each step the filter weights are updated by adding a correction term that brings the filter weights closer to the solution. The direction of the correction term points toward the steepest descent of \( \xi(n) \), see figure 2.4 for a graphical explanation. Mathematically the direction is denoted by the gradient. The gradient is the vector of partial derivatives of the MSE with respect to the weight of the filter vector \( w_n \). Since the gradient \( \nabla \xi(n) \) points in the direction of steepest ascent a change in sign to \( -\nabla \xi(n) \) gives the direction of steepest descent. To control how much the correction affects the old coefficient a variable \( \mu \) is introduced. This (positive) step size \( \mu \) multiplied with the negative gradient gives the expression of the update equation
\[ w_{n+1} = w_n - \mu \nabla \xi(n) \tag{2.9} \]

where the gradient vector of the MSE is
\[
\nabla \xi(n) = \nabla E\{e(n)^2\} \\
= E\{\nabla e(n)^2\} \\
= E\{2e(n)\nabla e(n)\} 
\]

and by using Eq. (2.8) it can be written as
\[ \nabla \xi(n) = E\{2e(n)x(n)\} \tag{2.13} \]

which leads to the following expression of the steepest descent algorithm [4]
\[ w_{n+1} = w_n - \mu E\{2e(n)x(n)\} \tag{2.14} \]

In the next section we take a closer look how to estimate \( E\{e(n)x(n)\} \) if the expectation is unknown.
Figure 2.4: Example of the Mean Square Error $\xi(n) = E\{e(n)^2\}$ of two filter weights is illustrated by a surface plot to the left and a contour plot to the right. The direction of the steepest descent points towards the minimum of the error surface (opposite to the gradient).

## 2.4 Least Mean Square

In the late 1950s the earliest works on adaptive filters started by a number of researchers who were working independently in theories and applications of such filters. Early developed was the least mean square (LMS) algorithm. Its simplicity and effectiveness made it especially useful to apply for the design of adaptive filters. The algorithm was presented by Widrow and Hoff in 1960 and is nowadays widely used in many different areas and applications with non stationary signals and environments [15].

The LMS algorithm can be seen as a special case of The Steepest Descent Adaptive Filter (Eq. (2.14)) since the term $\mu E\{e(n)x(n)\}$ is generally unknown it must be replaced with an estimate such as the sample mean

$$E\{e(n)x(n)\} = \frac{1}{L} \sum_{l=0}^{L-1} e(n-l)x(n-l) \tag{2.15}$$

by substitute Eq. (2.15) into Eq. (2.14) we have

$$w_{n+1} = w_n + \frac{2\mu}{L} \sum_{l=0}^{L-1} e(n-l)x(n-l) \tag{2.16}$$

The Eq. (2.16) can be simplified if we use a one-point sample mean ($L = 1$)

$$w_{n+1} = w_n + 2\mu e(n)x(n) \tag{2.17}$$
By these changes the weight vector update equation is written on a particularly simple form known as the LMS algorithm [15].

### 2.5 Normalized LMS

Normalized LMS (NLMS) is introduced to improve the step size $\mu$ used in the LMS algorithm. A too small step size result in slow convergence of the system, but a smaller error is generally given. A large step size makes the system converge faster, but the error is generally larger and stochastic behavior will occur and the adaption can even diverge. For a large or greatly varying input signal $x(n)$ the LMS algorithm become unstable. This behavior is known as gradient noise amplification. A solution to this problem is to use a variable step size that depends on the energy. Therefore in the NLMS algorithm the step size is adjusted by normalizing it with the energy of the input signal $x(n)$.

The energy of the input signal $x(n)$ can be expressed as

$$\|x(n)\|_2^2 = \sum_{k=(n-p)}^{n} |x(k)|^2 \quad (2.18)$$

the inverse energy multiplied by a coefficient $\beta$ becomes the step size for the NLMS algorithm

$$\mu(n) = \frac{\beta}{\|x(n)\|_2^2} \quad (2.19)$$

With the new $\mu$ inserted in the LMS weight vector update equation (Eq. (2.17)) gives the expression the for Normalized LMS (NLMS)

$$w_{n+1} = w_n + \frac{2\beta e(n)x(n)}{\|x(n)\|_2^2} \quad (2.20)$$

Although the NLMS algorithm solves the problem of gradient noise amplification, a similar problem occurs when $\|x(n)\|_2$ becomes too small. Therefore the NLMS algorithm is modified by adding a small positive number $\epsilon$ in the denominator.

$$w_{n+1} = w_n + \frac{2\beta e(n)x(n)}{\epsilon + \|x(n)\|_2^2} \quad (2.21)$$

The variable step size improves the stability and the convergence rate at little extra cost. The only extra computation to evaluate the normalizations term $\|x(n)\|_2^2$ involves two
s quar ing operations, one additions and one subtraction. Since the term can be calculated recursively

\[ \|x(n + 1)\|^2_{l^2} = \|x(n)\|^2_{l^2} + |x(n + 1)|^2 - |x(n - p)|^2 \]  

(2.22)

It can be shown that the NLMS algorithm converges in mean-square if the following convergence condition is satisfied [4].

\[ 0 < \mu < 2 \]  

(2.23)

Important to mention is that the convergence condition is considered in infinite precision. Since computers have finite precision we must have this in mind when selecting parameters. The fixed-point restrictions are described in more depth in chapter 2.8.

### 2.6 Leaky LMS

Introducing a leaky factor in the LMS algorithm will restrain the energy in the filter weights and stabilize the algorithm. The leaky factor is given by \( \nu = 1 - \mu \gamma \), where \( \gamma \) is a real positive constant chosen to fulfill \( 0 < \nu \leq 1 \). Thus the update of the adaptive filter with Leaky LMS is

\[ w_{n+1} = (1 - \mu \gamma)w_n + 2\mu e(n)x(n) \]  

(2.24)

The leaky LMS algorithm limits the growth of the filter and also reduces the numerical error in finite precision implementation. The disadvantage with the method is the small systematical error of the adaption which makes a perturbation of the filter [4].

### 2.7 Filtered-x LMS Algorithm

Filtered-x LMS (FxLMS) algorithm is an alternative form of the LMS algorithm for use when the adaptive filter is followed by a secondary path. The secondary path is denoted by \( S(z) \), see the block diagram in figure 2.5. In the ideal situation the error signal is found by the difference between the output from the primary path and the output from the adaptive filter, like the normal LMS.
In practice ANC-systems involves a secondary path including D/A-converter, amplifier, loudspeaker, the acoustic path from the loudspeaker to the error microphone etc. This can cause problems and the delay needs to be compensated. The most obvious solution is to place the inverse of the secondary path in series with the secondary path to reduce the impact. In general the secondary path is not a minimum phase systems with non existing inverse. A better solution to compensate for the delay is to filtering the input signal $x(n)$ to the adaptive control algorithm by an estimate of secondary path $\hat{S}(z)$

$$x'(n) = s'(n) * x(n)$$

where $x'(n)$ is the filtered signal at time $n$.

![Block diagram of Filtered-x LMS](image)

This modification is somewhat more complicated than the normal LMS algorithm but results in a improvement in performance by faster convergence, increased stability, and a more accurate adaptive filter $\hat{W}(z)$ [2].

### 2.8 Fixed-Point Arithmetic

A 16-bit processor stores numbers in a 16-bit integer format. This means that a 16 bits can represent $2^{16} = 65535$ values, typically the integers from $-32768$ to $-32767$.

The dynamic range (ratio of the largest and smallest positive number) for 16-bit quantization is

$$20 \cdot log_{10}(2^{15}) = 90\, \text{dB} \quad (2.26)$$
and the best signal-to-noise ratio (SNR) for a sinusoidal signal is

$$\frac{10 \cdot \log_{10}(2^{15}/2)}{1/12} = 98 \text{ dB}$$

(2.27)

The quantization noise is smaller for higher amplitude signals, therefore the power of signal should be as large as possible without overflow. Overflow occur when an arithmetic operation exceeds the capacity of the register used to the result. The addition of two 16-bit numbers will produce a 16 bit sum, therefore no roundoff error is introduced. The problem occurs when the sum fall outside the range of $-32768$ and $-32767$. The most effective technique to prevent overflow is to scale down the magnitude of signals before the overflow and then scale it back to the original level later in the process when the risk of overflow is removed. It need to be done carefully since scaling down increases the roundoff error. For example if a signal is halved the roundoff error is increased by about $6dB$ and is equivalent to losing 1-bit in representing the signal [7] [5] [16].

Because of the dynamic range limitations the designer has to apply scaling factors to prevent arithmetic overflow which can be a difficult and time consuming process. But in general fixed-point DSP-devices are cheaper and faster than floating-point DSP-devices since they use less silicon and have fewer external pins. So when choosing between fixed-point and floating-point DSP-devices it is important take in consideration the faster software development time compared to the extra cost of the DSP device itself.

### 2.9 Real-Time Constraints

The processor speed and the algorithm complexity determines the fastest possible sampling rate of the analog signals. A real-time DSP system have to fulfill the constraint that the signal processing time, $t_p$, must be less than the sampling period $T$ to complete the processing task before the new sample comes in

$$t_p < T$$

(2.28)

This constraint also restricts the highest sampling frequency that can be implemented on a DSP system. The lowest allowed sampling rate to avoid aliasing is determined by the Nyquists sampling theorem.
Theorem 2.9.1 (Nyquist sampling theorem). The sampling frequency $f_s$ should be at least twice the highest frequency $f_M$ contained in the signal.

\[ f_s \geq 2f_M \] (2.29)

By using this theorem the highest frequency that can be sampled in a real application is determined by

\[ f_M \leq \frac{f_s}{2} < \frac{1}{2f_p} \] (2.30)

Although the DSP systems are getting faster this is still a limiting factor. It is even more apparent when the cost of DSP chip is taken into account. So there is still a trade-off between costs and system performance at present [5].
Chapter 3

Methods

The most common methods used in an ANC-system are Feedforward and Feedback. The first method I implemented in the ANC-system was based on the Feedback method which only need one microphone instead of two as in the Feedforward method. After experiencing the restriction of the distance between the microphone and the speaker I decided to use Feedforward in this application. The procedure taken is general to first identify the secondary path and use this estimation in the Feedforward/Feedback algorithm.

3.1 Secondary Path Identification

The Secondary Path $S$ can be estimated off-line before the ANC-system is used. The coefficients of $\hat{S}$ are adapted to match the unknown transfer function $S$. This is made by a known sequence of training data as output from the speaker and measure the signal with a microphone (error microphone) inside the cabin. The estimation process is made by adaptively estimate the Secondary Path using the LMS algorithm Eq. (2.16). The input to the update equation is the training data sequence and the error (see figure 3.1). The error $e(n)$ is the difference between the true signal and the estimated signal and is calculated as:

$$e(n) = s(n) * x(n) - \hat{s}(n) * x(n)$$

(3.1)

The sequence of training data needs to contain frequencies over the entire frequency range of interest. A commonly used signal is white noise, which has a constant power over all frequencies. There is several other ways to create the training data. For example a bandlimited
signal with a flat spectral density over the frequency range of interest and zero elsewhere or a training signal consisting of a sinuswave with increasing or decreasing frequency. A solution to make the training signal more comfortable is to use music with rich frequency content in the desired range [3] [2].

![Block diagram of Secondary Path Identification](image)

**Figure 3.1: Block diagram of Secondary Path Identification**

### 3.1.1 Algorithm

1. Generate a sequence of training data $x(n)$.

2. Get the desired signal $d(n)$ from the sensor.

3. Calculate the output from the FIR filter $\hat{S}(z)$

   $$y(n) = \hat{s}(n) * x(n)$$  \hspace{1cm} (3.2)

4. Calculate the error signal

   $$e(n) = d(n) - y(n)$$  \hspace{1cm} (3.3)

5. Update the FIR filter $\hat{S}(z)$ by using the LMS algorithm

   $$\hat{s}_{n+1} = \hat{s}_n + \mu e(n)x(n)$$  \hspace{1cm} (3.4)

6. Repeat step 2 to 5 until the secondary path FIR filter $\hat{S}(z)$ converges to optimum solution i.e. until the MSE is minimized. After the identification of the secondary
path can the transfer function $\hat{S}(z)$ can be used in the ANC-system.

### 3.2 Feedforward ANC

The Feedforward ANC-System contains two microphones (sensors) and one speaker (secondary source). A reference microphone is placed near the noise source to sense the coherent signal before it propagates past the speaker. The error microphone is placed near the secondary speaker to create the desired “quiet zone”. The signals from the microphones are used as input for the adaption of the filter $W$ to determine the output to the secondary speaker. So the reference microphone measures the noise and the error microphone measure the performance of the ANC [2]. The block diagram of the Feedforward FxLMS ANC-system is shown in figure 3.2.

![Block diagram showing the principle of Feedforward sound cancellation](image)

**P - Primary path** The primary path $P$ is the acoustic transfer function from the reference sensor (near the noise source) to the error sensor inside the cabin.

**S - Secondary path** The secondary path $S$ is the transfer function from the $y(n)$ to $y'(n)$ and includes the digital-to-analog converter, speaker, acoustic path from speaker to error microphone, analog-to-digital converter etc.

**W - Adaptive filter** The objective of the adaptive filter $W$ is to minimize the noise measured by the error sensor.
3.2.1 Algorithm

1. Initiate the vector \( w_0 \) with zeros.

2. Get the signals from the reference sensor and the error sensor.

3. Calculate the output from the FIR filter \( W(z) \)

   \[
   y(n) = w_n(n) * x(n)
   \]

   (3.5)

4. Calculate the output from the FIR filter \( \hat{S}(z) \)

   \[
   x'(n) = \hat{s}(n) * x(n)
   \]

   (3.6)

5. Update the FIR filter \( W(z) \) by using the LMS algorithm

   \[
   w_{n+1} = w_n + \mu e(n)x'(n)
   \]

   (3.7)

   where \( e(n) = d(n) + y'(n) \) is the signal from the error sensor.

6. Send out the anti-noise signal \( y(n) \) from the speaker.

7. Repeat step 2 to 6 while the ANC-system is running.

3.3 Feedback ANC

The physical difference in Feedback compared to Feedforward is that no reference microphone is placed near the noise source. Instead is a reference signal estimated in the Feedback ANC-system based on the adaptive filter output signal and the error signal [2]. The block diagram of Feedback FxLMS ANC-system is shown in figure 3.3.

3.3.1 Algorithm

1. Initiate the vector \( w_0 \) with zeros.

2. Get the signal from the error sensor.
3.3. Feedback ANC

Figure 3.3: Block diagram showing the principle of Feedback sound cancellation

3. Estimate the primary signal as

\[ x(n) = e(n) - \hat{s}(n) * y(n) \]  \hspace{1cm} (3.8)

4. Calculate the output from the FIR filter \( W(z) \)

\[ y(n) = w_n(n) * x(n) \]  \hspace{1cm} (3.9)

5. Calculate the output from the FIR filter \( \hat{S}(z) \)

\[ x'(n) = \hat{s}(n) * x(n) \]  \hspace{1cm} (3.10)

6. Update the FIR filter \( W(z) \) by using the LMS algorithm

\[ w_{n+1} = w_n + \mu e(n)x'(n) \]  \hspace{1cm} (3.11)

where \( e(n) = d(n) + y'(n) \) is the signal from the error sensor.

7. Send out the anti-noise signal \( y(n) \) from the speaker.

8. Repeat step 2 to 7 while the ANC-system is running.
The Feedback ANC-system has one significant advantage i.e. no reference signal is required. But also several disadvantages like the noise reduction is limited over a restricted frequency range and require periodic noise. High frequencies can create instabilities, because of the possibility of positive feedback caused by phase difference. Another disadvantage is that the algorithm is not selective, so any signal will be reduced not just those correlated with the reference signal. The physical distance between the microphone and the speaker is restricted to a certain length depending on the frequencies of the noise, in the case of stochastic noise. If $c_0$ is the speed of sound and $L$ the distance between the loudspeaker and the microphone then the maximum frequency $f_M$ of operation is

$$f_M \ll \frac{c_0}{2L}$$

and usually are the upper frequency about one tenth of $c_0/2L$. Thus if the distance $L$ is about 50 cm and the speed of sound $c_0$ is 340 meters per second the upper frequency will be in practice 34 Hz [2]. So either the system will be restricted to reduce very low frequency noise or a short distance between the microphone and the speaker.

Having that in mind a robust system Feedback ANC-system can be developed in certain cases due to the predictable nature of narrow band signals. This algorithm is used in commercially available such as headphones and hearing protectors.
Chapter 4

Results

In this chapter are the results presented, including the results from the determination of the secondary path and the performance of the ANC-system. The data of the secondary path is collected in real time by the ANC-system inside a Komatsu forest machine. The performance of the ANC-system is measured by using recorded data from the operating forest machine. This signal is thereafter played outside a new cabin with the ANC-system installed. This is done both with the ANC-system turned off and turned on, so that it could be analyzed how the sound pressure levels are affected.

4.1 Secondary Path

The transfer function $P$ (primary path) gives information about what happens from the noise source outside the cabin to the microphone inside the cabin. Since this function is a physical filter it can not in general be exactly determined. Instead the transfer function is estimated by playing a white noise signal from a speaker close to the engine while a microphone placed in the roof of the cabin is recording the signal. The primary path is estimated in similar fashion.

The transfer function $S$ (secondary path) contains information about the path from the secondary source to the error sensor. Since this function is a physical filter it can not in general be exactly determined. Instead the transfer function is estimated by playing a white noise signal from a speaker and register the signal at the microphone. The speaker is placed at the floor of the cabin and the microphone in the roof. This setup together with the
ANC-system using the algorithm in section 3.1.1 gives the estimate of the secondary path. In figure ?? the time plot of the secondary path estimate is shown. The delay of the filter is about 0.025 s depends on the distance, speed of sound, speaker, DSP etc. The characteristic of the filter is a large peak followed by smaller peaks caused by echoes in the cabin.

Figure 4.1: Time plot of the Secondary path.
4.2 ANC performance with tonal noise

To evaluate how well the ANC-system is working a test was made by placing a speaker outside the cabin generating a tonal noise at 100 Hz. The input to the error microphone was recorded two times to see the difference between the ANC-system turned on and off. Time plot of the error signal in both cases are presented in figure 4.2. The SPL reduction measured with a decibel meter close to the microphone is 14 dB and the largest reduction occur at 100 Hz, 200 Hz and 300 Hz see figure 4.3.

Figure 4.2: Time plot of the error signal with a tonal noise of 100 Hz. The plot to the right is with ANC and to the left is without ANC.
Figure 4.3: Power Spectral Density Estimate of the error signal with a tonal noise of 100 Hz. The solid line is when the ANC-system is turned on and dashed line when turned off.
4.3 ANC performance with engine noise

With the same setup as with tonal noise the ANC-system was tested with recorded engine noise from an operating forest machine at 850 rpm. In the same way as for the tonal noise case, the error microphone was recorded two times to see the difference between the ANC-system turned on and off. Time plots of the error signal for both cases are presented in figure 4.4. The SPL reduction measured with a decibel meter close to the microphone was 11 dB and the largest reduction occur at 48 Hz and 96 Hz see figure 4.5.

![Figure 4.4: Time plot of the error signal at the engine speed 850 rpm. The plot to the right is with ANC and to the left is without ANC.](image-url)
Figure 4.5: Power Spectral Density Estimate of the error signal at the engine speed 850 rpm. The solid line is when the ANC-system is turned on and dashed line when it is turned off.
4.4 SPL measurements

The result of the SPL measured with the decibel meter is presented in table 4.1. The calculated SPL at the error sensor is showed in table 4.2.

Table 4.1: Sound pressure levels measured with the decibel meter

<table>
<thead>
<tr>
<th></th>
<th>Engine noise</th>
<th>Single tonal noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANC off</td>
<td>68</td>
<td>68</td>
</tr>
<tr>
<td>ANC on</td>
<td>57</td>
<td>54</td>
</tr>
<tr>
<td>ΔSPL</td>
<td>11</td>
<td>14</td>
</tr>
</tbody>
</table>

Table 4.2: Sound pressure levels measured with the error sensor

<table>
<thead>
<tr>
<th></th>
<th>Engine noise</th>
<th>Single tonal noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔSPL</td>
<td>14</td>
<td>29</td>
</tr>
</tbody>
</table>
Chapter 5

Conclusions and Future Work

In this thesis a prototype of the ANC-system was successfully implemented and evaluated. Better knowledge was achieved on adaptive filtering, fixed-point arithmetic’s, embedded systems as well as non-technical knowledge such as project arrangement and academic writings.

Both leaky FxLMS feedback and feedforward ANC methods were analyzed. The feedforward method was preferred because of its robustness and stability compared to the feedback control method. The systems complexity is a bit higher for the feedforward control but with more stability and less limitations in the physical setup makes it more useful in this case. The noise attenuation obtained using the leaky FxLMS feedforward was approximately 14 dB for a tonal 100 Hz signal and 11 dB using recorded engine noise from a forest machine at 850 rpm. The ANC-system could obtain good results even during speech inside the cabin because of its design to only attenuate low frequency noise.

There are several interesting ways to improve the ANC-system. The prototype consists of many dependent and independent variables that can be optimized even further by some systematical method such as ANOVA (analysis of variance). The physical arrangement of the system can be further adjusted to give an optimal performance combined with suitable installation design. One can use a tachometer or an accelerometer as a reference sensor instead a microphone for increased reliability. With a tachometer as a reference sensor one can ensure no acoustic feedback from the secondary source to the reference sensor. Inside the forest machine there will be complex wave propagation shapes at higher order modes that are difficult to control by a single loudspeaker. A solution to this problem can be a
multichannel system with a set of several loudspeakers and microphones. The algorithm can also be implemented in the frequency domain by using FFT (Fast Fourier Transform) to make the FIR filtering process more efficient for long filters. A virtual microphone can be used to attempt to override the problem with the distance between the microphone and the driver’s head. One possible solution is to integrate the ANC-system with the existing sound system inside the forest machine. By tracking and extracting the bass information from the audio signal one can increase the performance of the ANC-system and give the driver a much greater musical experience.
Chapter 6

Acknowledgments

I am very grateful for the opportunity to explore this field. It is a pleasure to thank those who made this Master's thesis possible.

First of all I will thank my supervisors, Magnus Berggren and Fredric Lindström, whose support and guidance from the beginning to the end enabled me to develop a deep understanding of the subject. Second, I want to thank Erik Fällman for being my examiner, and for all his feedback during the process. I would also like to send my regards to all co-workers at Limes Audio AB for making my time at the company enjoyable. Thanks to Peter Assarsson and all the employees at Komatsu Forest who helped me during the evaluation of the ANC-system.

At last, I would like to express my deepest gratitude to my wonderful family and friends for always supporting me and encouraging me.
References


Appendix A

Blackfin BF526

– Blackfin Processor Core with up to 400 MHz (800 MMACS) performance
– 2 dual-channel, full-duplex synchronous serial ports supporting 8 stereo I2S channels
– 12 peripheral DMA channels supporting one- and two-dimensional data transfers
– NAND Flash Controller with 8-Bit interface for commands, addresses and data.
– Ethernet 10/100 MII interface
– Memory controller providing glue-less connection to multiple banks of external SDRAM, SRAM, Flash, or ROM
– Low Standby Power; 1mA Deep Sleep, 50uA Hibernate
– 289-ball, 12x12 mm, 0.5 mm pitch mini-BGA (Commercial temperature range 0°C to +70°C)
– 208-ball, 17x17 mm, 0.8 mm pitch mini-BGA (Commercial temperature range 0°C to +70°C; Industrial
– temperature range −40°C to +85°C)

A.1 Tables

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enclosure system:</td>
<td>2-way bass-reflex system</td>
</tr>
<tr>
<td>Drivers:</td>
<td>LF 4&quot; cone speaker / HF 6/8&quot; soft dome tweeter</td>
</tr>
<tr>
<td>Freq. Response:</td>
<td>60Hz to 30kHz</td>
</tr>
<tr>
<td>Low Freq Cut-Off:</td>
<td>60Hz (-10dB)</td>
</tr>
<tr>
<td>High Freq Cut-Off:</td>
<td>22Hz (-10dB)</td>
</tr>
<tr>
<td>Output Power:</td>
<td>18W (LF), 18W (HF)</td>
</tr>
<tr>
<td>Input Terminals:</td>
<td>6mm TS Phone (unbalanced), RCA Pin (unbalanced)</td>
</tr>
<tr>
<td>Distortion:</td>
<td>Less than 0.1% T.H.D. @12W, 4-ohm 20Hz-20kHz</td>
</tr>
<tr>
<td>Signal to Noise Ratio:</td>
<td>18W (LF), 18W (HF)</td>
</tr>
<tr>
<td>Power Consumption:</td>
<td>40W</td>
</tr>
<tr>
<td>Physical Dimensions:</td>
<td>144mm (W) x 220mm (H) x 180mm (D)</td>
</tr>
<tr>
<td>Net Weight:</td>
<td>3.75kgs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Power:</td>
<td>140 W</td>
</tr>
<tr>
<td>RMS Power:</td>
<td>70 W (DIN 45324 @14.4 V)</td>
</tr>
<tr>
<td>I max/min:</td>
<td>10 A / &lt;3 mA</td>
</tr>
<tr>
<td>Phase:</td>
<td>0° / 180°</td>
</tr>
<tr>
<td>Frequency response:</td>
<td>30-120 Hz</td>
</tr>
<tr>
<td>Low pass filter:</td>
<td>40-120 Hz</td>
</tr>
<tr>
<td>Gain:</td>
<td>0.1V-8 V</td>
</tr>
<tr>
<td>Cone size:</td>
<td>200 mm / 8&quot;</td>
</tr>
<tr>
<td>Cone material:</td>
<td>Aluminium Compound</td>
</tr>
<tr>
<td>Size (WxHxD):</td>
<td>340 x 71/79 x 225 mm (13.4 x 2.8/3.1 x 8.9&quot;)</td>
</tr>
<tr>
<td>Weight:</td>
<td>4.3 kg</td>
</tr>
</tbody>
</table>
### Table A.3: Blaupunkt THb200A

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Power:</td>
<td>140 W</td>
</tr>
<tr>
<td>RMS Power:</td>
<td>70 W (DIN 45324 @14.4 V)</td>
</tr>
<tr>
<td>I max/min:</td>
<td>10 A / &lt;3 mA</td>
</tr>
<tr>
<td>Phase:</td>
<td>0° / 180°</td>
</tr>
<tr>
<td>Frequency response:</td>
<td>30-120 Hz</td>
</tr>
<tr>
<td>Low pass filter:</td>
<td>40-120 Hz</td>
</tr>
<tr>
<td>Gain:</td>
<td>0.1V-8 V</td>
</tr>
<tr>
<td>Cone size:</td>
<td>200 mm / 8&quot;</td>
</tr>
<tr>
<td>Cone material:</td>
<td>Aluminium Compound</td>
</tr>
<tr>
<td>Size (WxHxD):</td>
<td>340 x 71/79 x 225 mm (13.4 x 2.8/3.1 x 8.9&quot;)</td>
</tr>
<tr>
<td>Weight:</td>
<td>4.3 kg</td>
</tr>
</tbody>
</table>

### Table A.4: ECM-304BD

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:</td>
<td>electret/cardioid/omnidir.</td>
</tr>
<tr>
<td>Frequency range:</td>
<td>30-20 000Hz</td>
</tr>
<tr>
<td>Impedance:</td>
<td>600 Ohm/1,000 Ohm</td>
</tr>
<tr>
<td>Sensitivity:</td>
<td>7mV/Pa/1kHz</td>
</tr>
<tr>
<td>Max. SPL:</td>
<td>-</td>
</tr>
<tr>
<td>Power supply:</td>
<td>1.5V button cell (LR44)</td>
</tr>
<tr>
<td>Admiss. ambient temp.:</td>
<td>0-40 °C</td>
</tr>
<tr>
<td>Dimensions:</td>
<td>70x96x20mm</td>
</tr>
<tr>
<td>Weight:</td>
<td>300g</td>
</tr>
<tr>
<td>Connection:</td>
<td>3.5mm plug</td>
</tr>
</tbody>
</table>