Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
http://ens.ewi.tudelft.nl/

CAS-2018-00

M.Sc. Thesis

On speech enhancement in very low SNRs for smart speakers

Konstantinos Arsenios Sachos

Abstract

Human interaction with a smart speaker involves often distant automatic speech recognition (ASR). However, ASR is a rather cumbersome task at significantly high levels of noise. Most of commercial smart speakers in order to achieve high ASR accuracy they tend to reduce the playback signal once the preset keyword is detected. In an effort to dispose this function from the smart speaker, in this thesis a speech enhancement technique is considered in the front-end of the ASR system aiming at the suppression of the dominant noise component in the degraded speech signal. Having a priori knowledge on the playback signal renders adaptive filtering a well-suited speech technique. Therefore, the class of least mean squares (LMS) algorithms is studied and assessed. Among other techniques of this class the transform domain LMS (TDLMS), due to its inherent signal decorrelation properties, is shown to achieve the best performance in terms of noise suppression and improved speech intelligibility as well as word error rate. The results of this study correspond to a set of simulation incorporating real impulse responses measured in both an anechoic and a reverberant environment.

TU Delft

Faculty of Electrical Engineering, Mathematics and Computer Science

Delft University of Technology

On speech enhancement in very low SNRs for smart speakers

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Konstantinos Arsenios Sachos
born in Volos, Greece

This work was performed in:

Bang & Olufsen A/S
Department of Research & Development
Struer, Denmark

Delft University of Technology

Delft University of Technology
Department of
Microelectronics & Computer Engineering

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"On speech enhancement in very low SNRs for smart speakers"** by **Konstantinos Arsenios Sachos** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 8/10/18

Chairman: ____________________
prof.dr.ir. R. Heusdens

Advisor: ____________________
M.B. Møller, M.Sc., Bang and Olufsen

Committee Members: ____________________
dr.ir. R.C. Hendriks, TU Delft

____
dr. J.A. Martinez, TU Delft

____
dr. J. K. Nielsen, Bang and Olufsen & AAU

Abstract

Human interaction with a smart speaker involves often distant automatic speech recognition (ASR). However, ASR is a rather cumbersome task at significantly high levels of noise. Most of commercial smart speakers in order to achieve high ASR accuracy they tend to reduce the playback signal once the preset keyword is detected. In an effort to dispose this function from the smart speaker, in this thesis a speech enhancement technique is considered in the front-end of the ASR system aiming at the suppression of the dominant noise component in the degraded speech signal. Having a priori knowledge on the playback signal renders adaptive filtering a well-suited speech technique. Therefore, the class of least mean squares (LMS) algorithms is studied and assessed. Among other techniques of this class the transform domain LMS (TDLMS), due to its inherent signal decorrelation properties, is shown to achieve the best performance in terms of noise suppression and improved speech intelligibility as well as word error rate. The results of this study correspond to a set of simulation incorporating real impulse responses measured in both an anechoic and a reverberant environment.

Acknowledgments

The idea of pursuing a career abroad has been rather appealing to me since the very early years of my bachelor studies. Looking back at the beginning of my masters degree in TU Delft I remember being filled with excitement in taking my studies one step further. I could never imagine though that when signing up the first day I was actually signing up for a 2-year journey of invaluable knowledge and invaluable people entering my life.

Therefore, there is a number of people I would like to thank that have shaped my path during this last period. Among other scientific fields audio processing has always been the one that has attracted me the most. No wonder how I ended up following the *Signals and Systems* masters track. I thank all the academic personnel who paved the way of great learnings during the first year. In particular, I would like to thank my university supervisor Richard for being an inspiring professor as well as extremely helpful and cooperative during my whole masters degree and moreover because of him I conducted this thesis with Bang and Olufsen.

In turn, I would like to show my gratitude to the whole research group of Bang and Olufsen who created a friendly but at the same time very professional working environment. I wish mainly to express my appreciation to my supervisors Martin, Pablo and Jesper. Having to work and interact with these brilliant and innovative people made the experience of the last eight months rather enjoyable and priceless.

A huge thanks to my friends (both in Delft & Struer) that offered me an abundance of intense moments but most of all I thank them for our arguments that always led to a valuable lesson and a great deal of fun.

Finally, I choose to dedicate this thesis to my family who has supported me eversince and trusted my choices despite our different way of thinking. I will be always in dept to them for the things they have sacrificed so that I fulfill my dreams. A great share of this work is devoted to my sister, for her persistence and hard-working personality has been a driving force for whatever I do.

Konstantinos Arsenios Sachos
Delft, The Netherlands
8/10/18

Contents

List of Figures

List of Tables

1 Introduction

The use of smart audio devices as part of our autonomous assistance has increased rapidly over the last decade. At the same time, the technology involved in these devices has been advancing in a fast rate. As a result, smart speakers represent a massive and growing market with application to various ecosystems. One of the main functions of a smart loudspeaker is to execute one's voice command within the shortest possible period of time. This function is associated with the ability of the embedded automatic speech recognition (ASR) system to translate a speech signal into text. The key to a smart speaker's performance is the voice recognition technology that is used. Several approaches have been investigated for designing a robust ASR algorithm. However, ASR has proved to be a rather challenging task in very low signal-to-noise ratios (SNRs[1]) [1].

Most of the smart speakers in the market reduce the playback volume once a keyword has been detected. Although this reduction leads to an increase of the SNR, it could be rather annoying to the user if the keyword is misdetected. Therefore, the objective of this work is to support an ASR system with a front-end speech enhancement framework. The notion behind this approach is that feeding an ASR algorithm with an enhanced signal, would increase its performance in terms of improved WER. The metric of WER, being the most common performance measure for speech recognition systems, measures the difference between the transcribed word sequence and the reference one.

1.1 Thesis Objective

The main focus of this work is on speech enhancement algorithms that operate independently from the ASR system embedded in the smart speaker. There exist a great many signal processing techniques aiming at the reduction of both point

[1] Here, as SNR the ratio of speech power to noise power is defined.

sources and ambient noise which might degrade a speech signal. Of vital importance is the fact that the algorithm should operate in real time. This implies that computational complexity and robustness of the algorithm comprise crucial factors when designing such a framework. Multi-channel techniques such as beamforming and independent component analysis (ICA) have been studied intensely in the literature and have shown satisfactory performance under certain conditions. Moreover, adaptive filtering has been widely used to suppress the noise in corrupted speech signals. Being that explicit information on the interfering signal is used and due to ease of implementation, adaptive filtering is found to be a reasonable approach for the problem tackled in this work. A certain class of adaptive filtering algorithms are therefore studied and tested on suppressing the loudspeaker signal that corrupts the user's voice command.

Consequently the research question that this thesis poses is the following:

In what ways can an a speech enhancement framework support the front-end of a smart speaker's ASR system in extremely low SNR scenarios ?

1.2 Thesis Outline

The outline of the rest of the thesis is given below.

In Chapter 2 the problem formulation is being presented followed by the mathematical model of the investigated setup. Furthermore, Section 2.3 exhibits related previous work on speech enhancement. In particular, the concepts of beamforming, ICA and adaptive filtering are briefly described. By stating the limitations of the first two approaches to the proposed setup, strong argumentation follows making adaptive filtering the best fit for the problem at hand.

In Chapter 3 a certain class of adaptive filtering algorithms is derived and shortly reviewed. Starting from the same minimization criteria, namely the mean squared error (MSE), the properties of each algorithm are discussed. The chapter concludes with a number of simulations that compare the described algorithms.

Chapter 4 demonstrates the results of a Monte Carlo simulation. More explicitly, in Section 4.4 the findings of this work are discussed that reflect the plotted performance metrics.

The thesis concludes in Chapter 5 with the outlook of this work and a set of recommendations regarding potential future directions.

2 Preliminaries

This chapter focuses initially on establishing the problem formulation of this thesis that is subsequently supported by the mathematical model. The state of the art is moreover reported in the last section of this chapter justifying the selection of the speech enhancement approach that is elaborated in the next chapters.

2.1 Problem Formulation

The performance of an ASR system is highly dependent on the input SNR which is a function of the distance from each source to the microphone. In order to evaluate the expected input SNR, a ballpark example is sketched in Fig. 2.1 that corresponds to a real-time voice recognition scenario. The scenario is assumed as such by measuring the sound pressure level (SPL) at the point of interest, which is defined as

$$L_p = 10 \log_{10} \frac{p_{\text{rms}}^2}{p_{\text{ref}}^2}, \tag{2.1}$$

where p_{rms} is the root mean square value of the signal's sound pressure and p_{ref} is the reference sound pressure [2]. The sound pressure level is measured in dB_{SPL}. The correlation of the logarithmic scale with the subjective impression of loudness by the human ear justifies the fact that dB is chosen as a unit [3]. Moreover, in order to calculate the signal's SPL level at a point in r in free field given a point source at position r_0, the attenuation of the acoustic wave's sound pressure [2] is considered

$$p(\|r - r_0\|_2, \omega) = \frac{A}{\|r - r_0\|_2} \mathrm{e}^{-j\omega t}, \tag{2.2}$$

where A is the amplitude of the signal's sound pressure. In this scenario a single user is listening to music at $70\,\text{dB}_{\text{SPL}}$ at distance d_s from a smart loudspeaker in an anechoic environment. While music is playing, the user interacts with the loudspeaker via a voice command which is set at $70\,\text{dB}_{\text{SPL}}$ at $1\,\text{m}$ in front of him.

The selected sound pressure level of speech is with respect to the values reported in [4]. In contrast, the range of the pressure levels that people listen to music is large, hence an average value was chosen [5].

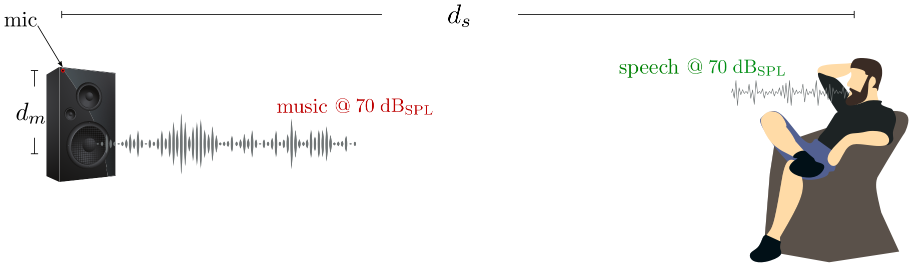

mic
d_s
d_m
music @ 70 $\mathrm{dB_{SPL}}$
speech @ 70 $\mathrm{dB_{SPL}}$

Figure 2.1: Scenario of a single user interacting with a smart loudspeaker. The $\mathrm{dB_{SPL}}$ of music is considered at the user's position, whereas the speech level is measured 1 m in front of the user.

The expected SNR at the microphone is calculated for various distances d_s of the user. In addition, the microphone is considered at two different distances d_m, with respect to the music source. One at 0.1 m and another at 0.2 m. The results are shown in Table 2.1. Of course, the further the microphone is placed from the music source, the better input SNR is obtained. However, since a loudspeaker has variable physical limitations in terms of size it is worth investigating both proposed distances.

d_m \ d_s	1 m	2 m	3 m	4 m	5 m
0.1 m	-21.3	-30.2	-39.3	-42.9	-45.4
0.2 m	-15.2	-24.6	-33.1	-37.4	-39.9

Table 2.1: Expected input SNRs in dB for various distances of the talker d_s and microphone d_m in meters.

2.2 System Model

The conceptual diagram of the problem that this thesis attempts to solve is depicted in Fig 2.2. The main elements of this setup are a smart loudspeaker with an embedded microphone and a target source. The audio (music) signal from the loudspeaker $u(n)$ comprises the interfering source, whereas the target source produces a desired speech signal $v(n)$. Both music and speech are convolved with the room impulse responses (RIRs) $\mathbf{h}_u(n)$ and $\mathbf{h}_v(n)$ respectively. Each RIR corresponds to the acoustic path from the source to the loudspeaker's microphone.

Finally, the microphone captures the mixed signal

$$\begin{aligned} d(n) &= v(n) * \mathbf{h_v}(n) + u(n) * \mathbf{h_u}(n) \\ &= s(n) + z(n). \end{aligned} \tag{2.3}$$

Provided that the music signal $u(n)$ is available, the goal is to estimate the speech signal $s(n)$ from $d(n)$ and feed it to the ASR system. If explicit information on the RIR $\mathbf{h}_u(n)$ was at hand, speech extraction would be a rather trivial problem to solve. Since this is not the case, a variety of techniques have been developed under a speech enhancement framework.

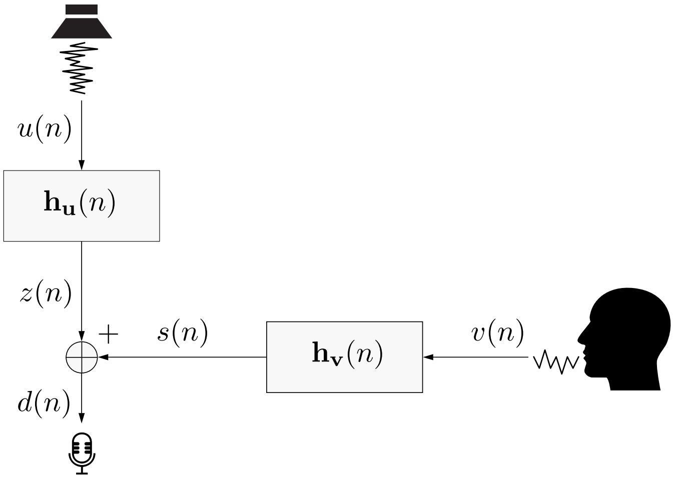


Figure 2.2: Conceptual diagram of the problem. The microphone is capturing the music $z(n)$ and the speech signal $s(n)$ after being convolved with the acoustic transfer functions $\mathbf{h_u}$ and $\mathbf{h_v}$ respectively.

2.2.1 Performance Measures

In order to assess the performance of the speech enhancement framework the necessary criteria need to be established. Depending on the application where speech needs to be isolated, different criteria are taken into account. At a first stage the output SNR of the framework is assessed. By denoting $\hat{s}(n)$ the estimated speech and $s(n)$ the clean speech signal, the output SNR is calculated

$$\mathrm{SNR_{dB}} = 10 \log_{10} \frac{|s(n)|^2}{|s(n) - \hat{s}(n)|^2}, \tag{2.4}$$

where $|\cdot|^2$ is the energy of the signal. However, in [6, 7] it is shown that the time-domain SNR measure does not correlate highly with either intelligibility or speech quality. Hence, for comparison purposes the intelligibility of the enhanced speech signal is calculated. For this task the short time objective intelligibility STOI measure [8] is considered. In contrast to other intelligibility models, STOI processes the signal in short frames (386 msec). This has proved to yield results equivalent to listening experiments [8]. In addition, in [9] among other perfomance metrics, STOI has shown to be the best predictor of ASR performance. The final assessment of the suggested framework is evaluated by the metric of word error rate (WER). For the calculation of WER we used DeepSpeech, an open source speech recognition algorithm developed by the machine learning group of Mozilla [10]. This speech-to-text engine uses a model trained by machine learning techniques based on the work described in [11].

2.3 State of the Art

In this section, three speech enhancement approaches are briefly outlined i.e. beamforming, independent component analysis (ICA) and adaptive filtering. First, the motivation behind each technique is demonstrated followed by the corresponding assumptions. Finally, the limitations of each technique with respect to our model and the available information are presented. The proposed speech algorithm for our model is based on adaptive filtering and hence this technique is regarded here last and is described in greater detail in the following chapter.

2.3.1 Beamforming

Beamforming is a spatial signal processing technique that uses multiple microphones to isolate target sources according to their position in space [12]. The simplest implementation of beamforming is the delay and sum beamformer (DSB). Supposing a single source is to be isolated, the idea of DSB is to align the output of the microphones by introducing the corresponding propagation delay of sound from the target source to each microphone. Subsequently, the microphone signals are summed and by assuming that the interfering signals arrive from various directions, the noise coefficients are added destructively. This results in the enhancement of the target signal and the suppression of the surrounding noise. Assuming that the microphone self-noise is uncorrelated across the microphones, the variance in the output of the beamformer is reduced by a factor of the number of the microphones M [13]. The DSB's major drawback is that the noise field is not taken into account. This implies that information about the structure and the direction of the interfering signals is not incorporated into the algorithm.

A widely used beamformer due to its robustness is the minimum variance distortionless response (MVDR) being is a special case of the linear contraint minimum variance (LCMV) beamformer [13]. As its name suggests, MVDR aims at preserving the magnitude and the phase of the target source while minimizing the variance in the output of the beamformer. [14]. Being a spatial filtering technique, a beamformer's performance is highly dependent on the amount of knowledge we have on the location(s) of the speech source(s), the acoustic transfer functions ATFs from the speech source(s) to the microphone array as well as the second order statistics of the interfering source(s). However, estimating the ATFs in a multi-microphone setup is a rather cumbersome task. A significant contribution to this open problem suggests the use of (relative transfer function (RTF))s instead which are easier to approximate [15].

The generalized sidelobe canceler (GSC) is another popular family of beamformers that has been studied extensively. This beamformer consists of two sub-systems, a fixed beamformer where usually DSB is chosen and an adaptive noise cancelling section. The second section is fed with an estimate of the noise profile. This estimate is the product of filtering out the desired signal(s) from the microphone measurements with a blocking matrix. In order for this setup to perform well, a good approximation of the RTFs from the desired signal(s) to the sensor array is required since they are the basis of the blocking matrix. More insight on the function of the adaptive noise cancelling system is given in the next chapter for it is the focus of this thesis.

Another set of beamformers that give the best linear solution in the mean squared sense go by the name of optimal linear multi-channel Wiener filter. It is shown in [16] that this type of beamformer is basically a concatenation of an MVDR beamformer and a post-processing single channel Wiener filter. In [17] the efficiency of the optimum linear filter, in terms of trade off of noise reduction and speech distortion, is extensively studied while different post-filtering techniques at the output of a beamformer are reported in [18, 19]

Although spatial array signal processing is used in numerous speech enhancement applications nowadays most of the beamforming techniques are unable to produce satisfying results in low SNRs [20]. More specifically, since the sensor's position is relatively close to the music source in our model, it would be rather challenging to steer the beamformer to the speech source even if exact information of its location is available. As a result, a beamformer would not be able to capture the desired spatial information of the setup, thus not producing the desired results.

2.3.2 Independent Component Analysis (ICA)

Independent component analysis (ICA) is a blind source separation (BSS) technique widely used to solve the well known cocktail-party problem [21]. While not used only for speech enhancement purposes, ICA has shown promising results in blindly separating sources that are mutually uncorrelated. Moreover, even though ICA is a multi-channel technique, it is mainly based on information-geometry theory rather than array signal processing [22]. In essence, ICA maximizes an objective measure of statistical independence among the target sources. The most popular implementations like Fast-ICA maximize the nongaussianity of the sources, through kurtosis or negentropy. ICA aims at the extraction of the target sources having information only on the mixed signals acquired from each sensor [23]. The desired signals that are obtained through this technique are decorrelated and a significant reduction in the higher-order statistical dependencies is achieved [22]. This is done by solving a linear model in the time domain of the form

$$\mathbf{x} = \mathbf{A}\,\mathbf{s}, \tag{2.5}$$

where $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_M \end{bmatrix}^T$ consists of the observations captured by the microphones and $\mathbf{s} = \begin{bmatrix} s_1 & s_2 & \dots & s_N \end{bmatrix}^T$ the decorrelated sources. Let $\mathbf{A}$ be an $M \times N$ matrix that mixes the components of interest s_i, where M is the number of the microphones and N the number of the extracted signals. By estimating $\mathbf{A}$, frequently referred to as the mixing matrix, its inverse is subsequently computed and the independent components are obtained. ICA is able to produce desired results under certain conditions and therefore is governed by the following fundamental assumptions:

- The sources $\mathbf{s}$ are statistically independent.
- At most one source can be assumed that follows a Gaussian distribution.
- Each target signal has zero mean $E\Big\{\mathbf{s}\Big\} = 0$ and unit variance $E\Big\{\mathbf{s}\mathbf{s}^{\mathbf{T}}\Big\} = \mathbf{I}$.
- The number of the sensors in the setup are at least as many as the sources.

Although the required non-gaussianity of the sources makes ICA a good candidate for speech and music signals [24], there exist some limitations in the use of this technique. It has been reported in [25] that the performance is severely affected when the available recorded signals have very low SNR which is mainly the case of the studied scenarios in this work. Another proof of the poor performance when noise is dominant in the measurement is a recent work in speech denoising [26]. In this study a single-channel ICA in combination with empirical mode decomposition (EMD) and Wavelet thresholding is used. The evaluation of this method showed that the attained gain is limited at 7 dB while the lowest input SNR assessed is at -6 dB. Another drawback of using blind ICA algorithms is that the target signals are recovered with arbitrary scaling and permutation which limits the performance of an ASR system [27]. Finally a preventing factor for the use of ICA techniques is that convergence is not guaranteed [28].

2.3.3 Adaptive Filtering

Adaptive filtering is a field of great interest in audio processing due its simplicity and robustness. Strong evidence of that are its numerous applications such as acoustic echo cancellation (AEC), system identification, channel equalization, adaptive noise cancellation adaptive noise cancellation (ANC) and many other. The motivation behind using an adaptive filter for an application can be summarized to the following reasons. First, an adaptive filter does not require the whole sequence of a signal to carry out the necessary computations. Instead, the output of the filter is produced with every incoming sample or after a defined chunk of samples. That makes adaptive filtering well-suited for real time applications since the processing delay introduced is small. Furthermore, compared to other techniques that process large data sequences, the adaptive filter updates its parameters whenever a new sample of the input signal arrives thus not being so demanding in terms of memory usage. Moreover, an adaptive filer is easy to implement both from a software and hardware perspective [29]. Finally, adaptive filters are superior to other signal processing techniques in the sense that they can track changes both in the signal statistics and in the involved environment [30].

The setup of this study (Fig. 2.2) can easily be translated to an ANC problem. Most ANC systems consist of two inputs and one output. More specifically, the microphone signal $d(n)$ which contains speech corrupted with the uncorrelated music signal, comprises the primary input of the system. Another sensor, receives the reference music signal $u(n)$ which is assumed to be uncorrelated with the speech signal $s(n)$ but correlated with the noise component of $d(n)$. This secondary input is also the input of the adaptive filter. The digital filter's output is then subtracted from the primary input producing the output of the ANC system. In fact, the adaptive filter is adjusting its parameters by minimizing cost functions of the estimated error signal so that its output is as close as possible to the noise profile of the primary input. In our scenario, the output of the adaptive filter gets subtracted form the microphone signal yielding to an estimate of the speech signal.

Although, the concept of ANC and its application to our model is going to be extensively explained in the following chapter, it is essential to argue why adaptive filtering surpasses beamforming and ICA in solving the problem at hand. First, as already mentioned the incorporated filter can adapt to the conditions of the environment which makes it a very powerful technique since most the real systems are time-varying. Furthermore, in contrast to spatial techniques like beamforming, no explicit information on the location of the noise or the target source is needed. Last but not least, adaptive filtering exploits directly the information about the noise that corrupts the signal of interest. Therefore, since in our model the loudspeaker signal is available, there is no need in using preprocessing techniques on the microphone signal to get an estimate of the noise component.

However, similar to any other speech enhancement technique, adaptive filtering is governed by certain limitations. The convergence speed, being one of the main characteristic performance measures of an adaptive algorithm, measures how fast the filter adapts its coefficients to the desired state. Real time applications such as ANC require usually fast convergence [29, 30]. Adaptive filtering is an iterative method which implies the existence of a step size that controls the adaptation of the filter's parameters. As a result, the choice of the step size has a direct impact on the convergence speed of the algorithm. Several methods have been implemented to address this problem by either making the step size time varying [31–34] or by deriving an optimal step size based on various criteria [35–37]. In addition, the length of the adaptive filter is another factor that affects the convergence of the iterative process. For applications where the impulse response of a room needs to be modeled several hundred of filter taps are required. The longer the filter is, the slower the convergence of the filter coefficients [38]. In search for an ideal filter length in AEC applications, recent studies have used different optimization frameworks [39, 40]. Finally, a certain class of adaptive algorithms based on the minimization of the mean squared error (MSE) have received a lot of attention due to their robustness and ease of implementation which reduces the computational cost. Nonetheless, convergence on this type of algorithms is relatively slow for colored input signals [41]. Since music and speech are considered as such a great emphasis is given on techniques that attempt to resolve this issue.

3 Adaptive Filtering

In this chapter the fundamentals of adaptive filtering are presented. First, a closed form solution of the Wiener filter is derived while showing that is optimal in the mean squared sense. Subsequently, the motivation for approximating the Wiener solution iteratively is given. Translating the problem of this thesis into a noise cancellation scenario it is proved both mathematically and geometrically that under certain assumptions ANC can be considered a speech enhancement technique. The family of least mean squares algorithms is next introduced along with their convergence properties. Finally, a set of simulations highlights the superiority of the transform domain adaptive filter in terms of convergence speed in the case of highly correlated input signals like music. Since in the current work the involved signals are audio signals all the derived equations are real valued.

3.1 Wiener Filter

The scheme of adaptive filtering is demonstrated in Fig. 3.1. The model consists of an M-tap filter [30] denoted as $\mathbf{w}(n) = \begin{bmatrix} w_1(n) & w_2(n) & \dots & w_M(n) \end{bmatrix}^T$, the input of the filter $u(n)$, the desired signal $d(n)$ and finally the error signal $e(n)$ which controls the adaptation process. By $(\cdot)^T$ we refer to the *transposed* operator.

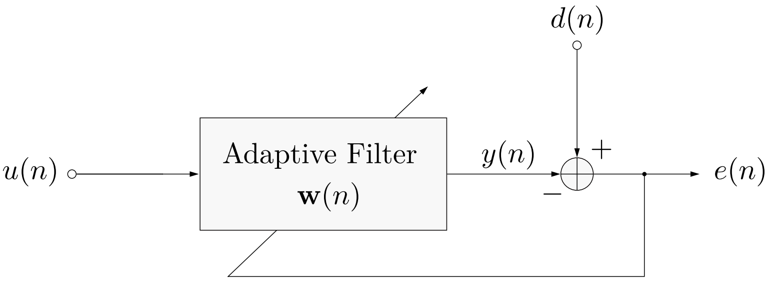


Figure 3.1: Block diagram of the adaptive filter.

Several studies using infinite impulse response (IIR) filters in adaptive filtering have been conducted in [42–44] due to their less complex design, in terms of required memory and calculations [45]. However, finite impulse response (FIR) filters are considered in this framework due to their implementation simplicity and their invulnerability to finite word length effects [46].

Since adaptive filtering is an iterative method, the input vector in every iteration comprises the past M samples of the reference signal $u(n)$, i.e. $\mathbf{u}(n) = \Big[u(n) \quad u(n-1) \quad \dots \quad u(n-M+1)\Big]^T$. The output of the filter can then be written as a linear convolution of the input vector and the filter weights

$$y(n) = \sum_{k=1}^{M} w_k(n)u(n-k). \tag{3.1}$$

The filter taps are updated by minimizing an objective function which in most of the cases is the mean squared error (MSE)

$$\xi(n) = E\Big\{e^2(n)\Big\} = E\Big\{\Big(d(n) - y(n)\Big)^2\Big\} = E\Big\{\Big(d(n) - \mathbf{w}^T(n)\mathbf{u}(n)\Big)^2\Big\}. \tag{3.2}$$

Assuming that $u(n)$ and $d(n)$ are wide-sense stationary (WSS) processes makes the filter coefficients time-invariant [30] and the MSE can be written as follows

$$\begin{aligned}\xi(n) &= E\Big\{d^2(n)\Big\} - 2\mathbf{w}^T E\Big\{d(n)\mathbf{u(n)}\Big\} + \mathbf{w}^T E\Big\{\mathbf{u}(n)\mathbf{u}^T(n)\Big\}\mathbf{w} \\ &= \sigma_d^2 - 2\mathbf{w}^T\mathbf{r_{du}} + \mathbf{w}^T\mathbf{R_u}\mathbf{w},\end{aligned} \tag{3.3}$$

where σ_d^2 is the variance of the desired signal $d(n)$, $\mathbf{r_{du}}$ is the cross-correlation between the desired and the input signal and $\mathbf{R_u}$ is the autocorrelation matrix of the regressor. Assuming nonsingularity on $\mathbf{R_u}$ and setting the gradient of the MSE to zero gives the optimal weight vector $\mathbf{w}_o$, also called the Wiener solution

$$\mathbf{w}_o = \mathbf{R_u}^{-1}\mathbf{r_{du}}. \tag{3.4}$$

The minimum MSE is obtained by evaluating the objective function at the optimal solution $\mathbf{w}_o$

$$\begin{aligned}\xi_{\min} &= \sigma_d^2 - 2\mathbf{w}_o^T\mathbf{r_{du}} + \mathbf{w}_o^T\mathbf{R_u}\mathbf{R_u}^{-1}\mathbf{r_{du}} \\ &= \sigma_d^2 - \mathbf{w}_o^T\mathbf{r_{du}}.\end{aligned} \tag{3.5}$$

Furthermore, an important property of the Wiener Filter is shown by examining the MSE surface at the optimal solution. The gradient of the error surface expressed as

$$\nabla\xi(n) = \frac{\partial E\Big\{e^2(n)\Big\}}{\partial\mathbf{w}} = E\left\{2e(n)\frac{\partial e(n)}{\partial\mathbf{w}}\right\} = -2E\Big\{e(n)\mathbf{u}(n)\Big\} \tag{3.6}$$

is equal to the zero vector for $\mathbf{w} = \mathbf{w}_o$ yielding to

$$E\Big\{e(n)\mathbf{u}(n)\Big\} = \mathbf{0} \tag{3.7}$$

This the well-known orthogonality principle [30] which implies that the error and the input signal are orthogonal to each other and, provided that the processes are zero-mean, they are also uncorrelated. This fundamental property is established in the 3.1.1.2 and signifies the assumption that the music and the speech vector are orthogonal. Moreover, it constitutes the basis for suppressing the noise in an adaptive noise cancelling scheme, a concept described in the following section.

3.1.1 Adaptive Noise Cancelling

Adaptive noise cancellation (ANC) is one of the most popular applications of adaptive filtering firstly introduced by Widrow *et. al* in [47]. The idea behind ANC is to design a system which will cancel the noise corrupting the signal of interest. The link between this notion and the objective of this thesis is quite apparent. The block diagram in Fig. 3.2 shows a speech enhancement framework using an adaptive noise canceler. A digital filter along with two inputs and a single output comprise the main elements of an ANC system. Adjusting the system to our model, let $s(n)$ be the speech signal and $u(n)$ the music signal being played from the loudspeaker. In addition, let $z(n)$ be the music component that arrives to the microphone after being convolved with the unknown transfer function $\mathbf{h_u}(n)$. The speech signal mixed with this music component constitute the primary input of

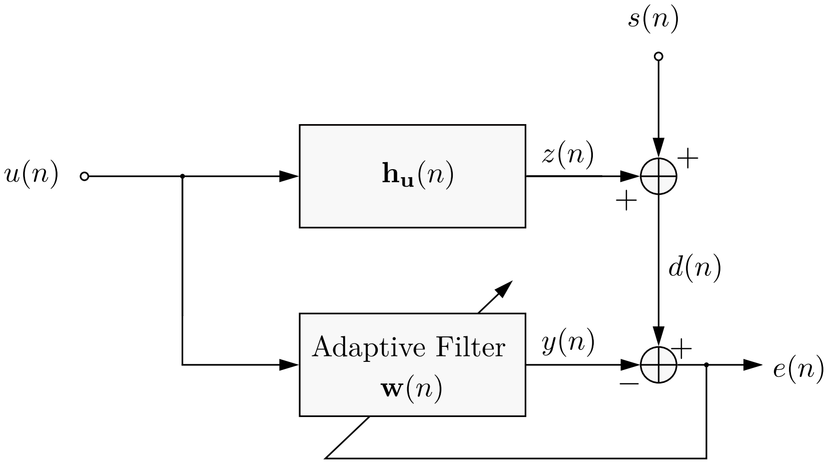


Figure 3.2: Block diagram of an adaptive noise canceler (ANC). The adaptive filter provides the best linear map of the input music signal $u(n)$ to the music component $z(n)$ of the microphone signal $d(n)$. The output of the ANC is eventually an approximation of the speech signal $s(n)$.

the system, i.e. $d(n) = s(n) + z(n)$. The music signal $u(n)$ provides the secondary input to the canceller which is also the input of the adaptive filter. The digital filter adapts its coefficients so that $u(n)$ is mapped to the correlated music component $z(n)$ of the primary input signal. Ultimately, ANC aims at minimizing the output of the system $e(n) = d(n) - y(n)$ which in this setup eventually yields to an estimate of the speech signal. This notion is justified first mathematically and then geometrically in the following sections.

3.1.1.1 Mathematical Interpretation

In order to prove that the output of the described system is approximating the speech signal $s(n)$, the following assumptions are made:

(a) $s(n),\ u(n),\ y(n),\ z(n)$ are zero-mean stationary processes.

(b) the speech signal $s(n)$ is uncorrelated with the interfering music component $z(n)$.

(c) the reference music signal $u(n)$ is correlated with the music component $z(n)$ of the primary input signal $d(n)$ but uncorrelated with the speech signal $s(n)$.

The output of the system can then be written as follows

$$e(n) = d(n) - y(n) = s(n) + z(n) - y(n). \tag{3.8}$$

Taking the square and subsequently the expectation on both sides of Eq. (3.8) gives

$$E\Big\{e^2(n)\Big\} = E\Big\{s^2(n)\Big\} + E\Big\{\Big(z(n) - y(n)\Big)^2\Big\} + 2E\Big\{\cancelto{0}{s(n)\Big(z(n) - y(n)\Big)}\Big\}. \tag{3.9}$$

The term that goes to zero in the above equation relates back to the assumptions (b) and (c). Therefore minimizing the total output power $E\Big\{e^2(n)\Big\}$ with respect to the filter coefficients is equivalent to minimizing $E\Big\{\Big(z(n) - y(n)\Big)^2\Big\}$, making the output of the adaptive filter $y(n)$ the best fit to the captured music signal $z(n)$ in the least squares sense [47]. The Wiener solution is the optimal least squares solution of the above minimization problem and when it is attained the output signal-to-noise ratio is maximized. That implies that $E\Big\{\Big(z(n) - y(n)\Big)^2\Big\} = 0$ and therefore the filter's output is a replica of the music component $z(n)$. Finally, from Eq. (3.8) it is obvious that the error $e(n)$ of the ANC system becomes equal to the speech signal $s(n)$.

3.1.1.2 Geometrical Interpretation

Here, the result of Eq. (3.7) is demonstrated from a geometrical perspective. Namely, considering our model, it is shown that the Wiener solution $\mathbf{w}_o$ suggests that the speech and the music vectors are orthogonal to each other. The geometrical illustration of the orthogonality principle is depicted in Fig. 3.3. Let $\mathbf{U} = \begin{bmatrix}\mathbf{u}(n) & u(n+1) & \dots & \mathbf{u}(n+M-1)\end{bmatrix}^T$ be the convolution matrix defined by the sequential vectors of the input signal $u(n)$. In addition, let $\mathbf{z} = \mathbf{U}\mathbf{w}_o$ be the music component which mixed with speech $\mathbf{s}$ define the noisy speech vector $\mathbf{d} = \mathbf{s} + \mathbf{z}$. The objective of the minimization problem

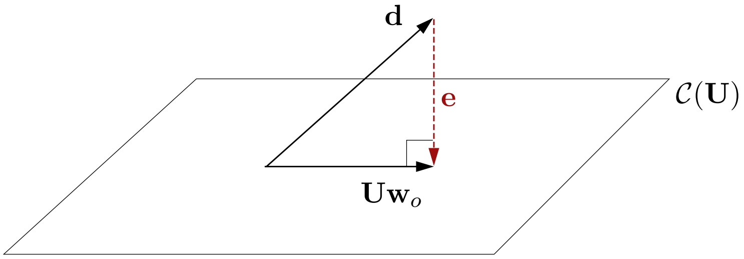


Figure 3.3: Orthogonality principle. The optimal least squares solution implies that the error $\mathbf{e}$ is orthogonal to all row vectors of the music matrix $\mathbf{U}$. Assuming speech and music are uncorrelated, the error vector $\mathbf{e}$ corresponds to the speech component $\mathbf{s}$ of the observation vector $\mathbf{d}$.

$$\min_{\mathbf{w}} \qquad \|\mathbf{d} - \mathbf{U}\mathbf{w}\|^2 \tag{3.10}$$

is to find the vector $\mathbf{w}$ in the column-space $\mathcal{C}(\mathbf{U})$ that is closest to the vector $\mathbf{d}$ in the Euclidean norm sense [41]. The solution is of course the projection of $\mathbf{d}$ onto the hyperplane defined by $\mathcal{C}(\mathbf{U})$, which implies that the error vector $\mathbf{e}$ is orthogonal to the space spanned by the $\mathbf{U}\mathbf{w}_o$ vector. Also, it is assumed that

$$\mathbf{s} \in \mathcal{N}(\mathbf{U}^T) \quad \text{and} \quad \mathbf{z} \in \mathcal{C}(\mathbf{U}),$$

where $\mathcal{N}(\mathbf{U}^T)$ is the null-space of $\mathbf{U}^T$. As a result, the residual error $\mathbf{e}$ is equivalent to the speech vector $\mathbf{s}$.

Having proved that under certain assumptions the output of the ANC system is equal to the speech signal, the next focus is on the iterative schemes of the incorporated adaptive filter. A certain class of algorithms is therefore presented along with their properties and limitations.

3.2 Least Mean Squares

In real time applications such as ANC the whole realization of the signals is usually not available and therefore the closed form solution of Eq. (3.4) cannot be used. Instead, an iterative scheme is preferred which starts with an initial guess of the optimal solution. Most of the adaptive schemes are based on the method of steepest descent [41] which is described by

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \frac{1}{2}\mu\,\nabla\xi(n) = \mathbf{w}(n) + \mu\,E\Big\{e(n)\mathbf{u}(n)\Big\}, \tag{3.11}$$

where μ is the step size that controls how fast the filter taps converge to the optimal solution. The steepest descent searches for the minimum of the objective function $\xi(n)$ in the opposite direction of its gradient. Providing that $\xi(n)$ is a convex function any minimum attained by this scheme is also a global minimum of the objective function. The minimization of the MSE depends on the statistics of the involved signals on which little or no information is usually available. As a result, different approximations of Eq. (3.6) have been investigated which lead to the development of different methods. The most popular of them uses an instantaneous estimate and goes by the name of LMS. The LMS block diagramm is depicted in Fig. 3.4. The input signal of the adaptive filter is processed in a tapped delay line (TDL) form and thus the update equation of the LMS algorithm is given by

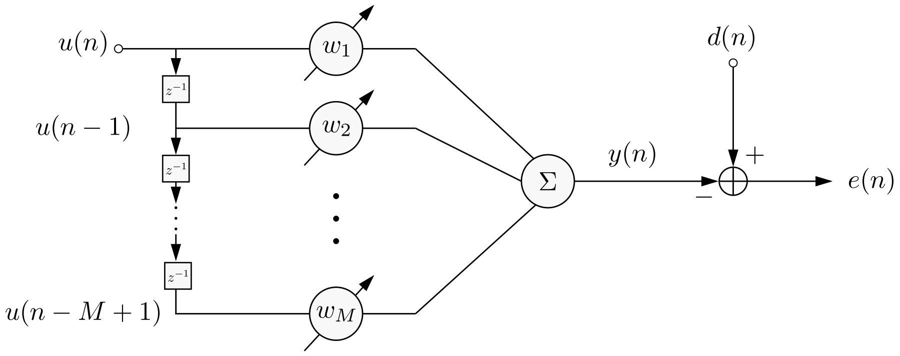


Figure 3.4: Block diagram of the LMS adaptive filter.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\,e(n)\,\mathbf{u}(n). \tag{3.12}$$

The LMS algorithm is widely used in many real time applications mainly due to its low computational complexity and low sensitivity to quantization errors [41]. However, the main challenge of the iterative schemes used in real-time applications is attaining the solution within few iterations. That implies that the convergence speed of the algorithm plays a significant role in the selection of the algorithm and is investigated first for the case of LMS.

3.2.1 Convergence of the LMS algorithm

The convergence of the LMS algorithm depends highly on the step size μ. By inspecting the eigenvalues, λ_k of the auto-correlation matrix $\mathbf{R_u}$, it is proven in [48] that the filter converges to the Wiener solution if $|1 - \mu\lambda_k| < 1$, for every k. This is satisfied by choosing the step size according to the following inequality

$$0 < \mu < \frac{2}{\lambda_{\max}}, \tag{3.13}$$

where $\lambda_{\max}$ corresponds to the maximum eigenvalue of $\mathbf{R_u}$. The above inequality indicates that $\lambda_{\max}$ sets a limit on the learning rate of the algorithm while keeping the adaptation process stable. On the other hand, the smallest eigenvalue defines how slow the filter coefficients adapt to the desired values [49]. Especially in highly colored signals $\lambda_{\min}$ can get close to zero resulting in very slow converging rates. In addition, non-stationary signals like music and speech present significant changes in the variance across time [50]. Abrupt changes in the energy of the filter's input signal have been dealt with a the normalized LMS algoritm which is introduced in the following section. The notion of this method is to control the adaption process with a normalization term that compensates for large variations in the energy of the input signal.

3.3 Normalized LMS

The NLMS is one of the fundamental algorithms in the LMS family and is of great interest due to its ability to deal better with slowly varying nonstationary input signals. The NLMS algorithm is derived from the Gauss-Newton method where the use of second order statistics reduces the algorithms dependence on the input power spectrum leading to improvement in the convergence rate of the algorithm [51]. The Gauss-Newton update scheme for an FIR adaptive filter is given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \frac{1}{2}\,\mu\left[\nabla^2\xi(n)\right]^{-1}\nabla\xi(n). \tag{3.14}$$

The difference between the traditional LMS and NLMS lies on the search direction where the latter method incorporates the Hessian of the minimized cost function. For the case of a stationary input, the Hessian is equal to $\nabla^2\xi(n) = E\Big\{\mathbf{u}(n)\mathbf{u}(n)^T\Big\} = \mathbf{R_u}$. Therefore by approximating again the gradient with an instantaneous estimate the update equation becomes

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\, e(n)\, \mathbf{\hat{R}_u^{-1}}(n)\, \mathbf{u}(n). \tag{3.15}$$

Usually an instantaneous approximation of the autocorrelation matrix is chosen $\hat{\mathbf{R}}_{\mathbf{u}}(n) = ||\mathbf{u}(n)||^2$ which gives the final form of the NLMS update rule

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{\mathbf{u}(n)}{\epsilon + ||\mathbf{u}(n)||^2} e(n), \tag{3.16}$$

where the additional parameter ϵ corresponds to a regularization term introduced to compensate for noise amplification in the case of a very small $||\mathbf{u}(n)||$. The inverse autocorrelation matrix acts as a normalization factor that compensates for the large fluctuations in the input signal. The choice of NLMS is quite intuitive when dealing with signals like music which are usually governed by large variations in the power levels.

3.3.1 Convergence of the NLMS algorithm

In addition to the bounds of the step size given in Eq. (3.13) it is shown in [48] that the LMS algorithm converges in the mean-square if

$$0 < \mu < \frac{2}{\mathrm{tr}(\mathbf{R}_{\mathbf{u}})}, \tag{3.17}$$

where $\mathrm{tr}(\mathbf{R}_{\mathbf{u}})$ is the trace of the input autocorrelation matrix. Furthermore for a stationary process $u(n)$ it holds

$$\mathrm{tr}(\mathbf{R}_{\mathbf{u}}) = M r_u(0) = M E\Big\{u^2(n)\Big\},$$

where $E\Big\{|u(n)|^2\Big\}$ is the power of $u(n)$, and could be estimated using an average such as

$$E\Big\{u^2(n)\Big\} \approx \frac{1}{M} \sum_{k=0}^{M-1} u^2(n-k).$$

As a result the inequality in (3.17) can be written as follows

$$0 < \mu < \frac{2}{||\mathbf{u}(n)||^2}. \tag{3.18}$$

By rearranging the above inequality, the LMS algorithm can be rewritten incorporating the time variant step size $\tilde{\mu}(n) = \mu \,/\, \Big(\epsilon + ||\mathbf{u}(n)||^2\Big)$ which if substituted yields the update equation of the normalized LMS in Eq. (3.16). The step size μ is now data independent since NLMS converges to the mean square for $0 < \mu < 2$.

Although NLMS has shown better overall performance in terms of convergence speed for non-stationary signals, the learning rate of the algorithm is relatively slow for colored inputs. The class of transform domain adaptive filters (TDAF) has been developed to resolve this issue and is described in the following section.

3.4 Transform Domain LMS

The concept of implementing a frequency domain realization of the traditional LMS adaptive filter, was first introduced by Dentino *et. al* [52], showing significant reduction in the computational complexity of the algorithm in terms of number of multiplications. Furthermore, this work gave rise to other investigations [53], [49] which proved that a substantial gain in the speed of the algorithm's adaptation can also be achieved. Thus, the family of TDAFs consists of two main categories, one that focuses on numerical efficiency and another focusing on the improvement of the convergence behavior. Here, the latter category is of interest since the nature of the input music signal in our framework causes an inherent problem in the convergence of the adaptive filter.

The performance of certain real-time applications, like ANC is highly dependent on the convergence speed of the adaptive scheme. In turn, the convergence speed of the filter coefficients is directly related to the second-order statistics in the input signal and more specifically to the eigenvalue spread of its covariance matrix, $\mathbf{R_u} = E\{\mathbf{u}(n)\mathbf{u}^{\mathrm{T}}(n)\}$ [41]. Also encountered as condition number in the literature, the eigenvalue spread is defined as

$$\kappa(\mathbf{R_u}) = \frac{\lambda_{\max}}{\lambda_{\min}},$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the largest and smallest eigenvalues of $\mathbf{R_u}$ respectively. The conditioning of $\mathbf{R_u}$ is determined by the value of this quantity. A large value of the condition number of $\mathbf{R_u}$ implies high correlation of the input signal which results in slow convergence of the adaptive filter. Hence, a pre-whitening technique is needed that would decompose the colored input signal, in our case the music signal, into orthogonal components and thereby reduce the eigenvalue spread. Next, the use of orthogonal transforms as a decorrelating technique is being introduced and the performance of various transformations is briefly reported.

3.4.1 Orthogonal Transforms

The Karhunen Loéve transform (KLT) is an optimal orthogonalizing technique in terms of signal decorrelation. Let $\mathbf{q}_i$ be the eigenvector associated with the *i-th* eigenvalue of the input correlation matrix $\mathbf{R_u}$. The transformed elements after applying the KLT would be of the form

$$x_i(n) = \mathbf{q}_i^T\mathbf{u}(n), \qquad i = 0, \ldots, M-1.$$

As mentioned in section 3.3, the Hessian in the Gauss-Newton's recursion corresponds to the autocorrelation maix of the input signal $\mathbf{u}(\mathbf{n})$. Since the eigenvectors are orthogonal to each other by construction, the transformed vector

$\mathbf{x}(n) = \begin{bmatrix} x_0(n) & x_1(n) & \dots & x_{M-1}(n) \end{bmatrix}^T$ becomes uncorrelated and its autocorrelation matrix would be of the form

$$\mathbf{R}_{\mathbf{x}} = E\Big\{\mathbf{x}(n)\mathbf{x}^T(n)\Big\} = \begin{bmatrix} \lambda_0 & & \\ & \ddots & \\ & & \lambda_{M-1} \end{bmatrix},$$

where λ_i are the eigenvalues of $\mathbf{R}_{\mathbf{x}}$. However, since the KLT transformation is signal dependent, its computational complexity is a prohibiting factor in real-time applications such as ANC. Instead a sub-optimal transform is often chosen. It is worth mentioning that the recursive least squares (RLS) is a class of algorithms that attain near optimal performance [54]. These algorithms use a time-varying estimate of $\mathbf{R}_{\mathbf{u}}$ to whiten the regressor. The inversion of this estimate though is rather expensive and hence RLS algorithms are not preferred in time-critical applications although their performance might be superior to the conventional LMS.

The need of a computationally efficient orthogonal transform led to the development of the transform domain LMS (TDLMS) based on fixed unitary orthogonal transforms. Among others, the discrete Fourier transform (DFT) and the discrete cosine transform (DCT) were evaluated first by Narayan in [53]. It was shown that the DCT achieves a significant reduction of the condition number of the input autocorrelation matrix which in turn improves the convergence speed of the adaptive filter. Moreover, DCT is a real-valued transform which makes it computationally more efficient to use when dealing with real-valued signals, such as audio signals [55]. In addition, in [56] the use of the Walsh-Hadamard transform (WHT), the discrete Hartley transform (DHT) and the powers-of-2 transform (PO2) transform were tested alongside the DCT, where again the latter outperformed the rest by significantly reducing the eigenvalue spread of the input correlation matrix. As a result, the DCT is considered and evaluated in this work.

3.4.2 Orthogonalization and Power Normalization

The TDLMS adaptive scheme is shown in Fig. 3.5. Compared to the traditional LMS, TDLMS involves two additional operations, both applied to the input signal. First, the orthogonalization of the regressor is carried out, followed by a power normalization step. At the first stage, a unitary matrix $\mathbf{T}$ is applied to the M past input samples of the input $\mathbf{u}(n) = \begin{bmatrix} u(n) & u(n-1) & \dots & u(n-M+1) \end{bmatrix}^T$. The transformation matrix $\mathbf{T}$ is a square $M \times M$ fixed matrix consisting of orthonormal vectors, where M is the filter length. The transformed vector $\mathbf{x}(n)$ is then defined as

$$\mathbf{x}(n) = \begin{bmatrix} x_1(n) & x_2(n) & \dots & x_M(n) \end{bmatrix}^T = \mathbf{T}\,\mathbf{u}(n) \tag{3.19}$$

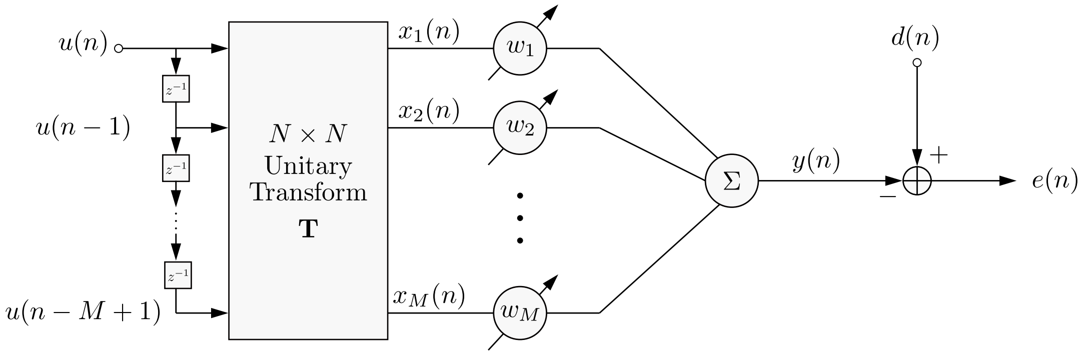


Figure 3.5: Block diagram of the transform domain LMS adaptive filter.

and the adaptive weight vector as

$$\mathbf{w}(n) = \begin{bmatrix} w_1(n) & w_2(n) & \ldots & w_M(n) \end{bmatrix}^T. \tag{3.20}$$

Hence, the output of the adaptive filter and the instantaneous error are given by

$$y(n) = \mathbf{w}(n)^T \mathbf{x}(n) = \mathbf{w}(n)^T \mathbf{T}\,\mathbf{u}(n) \tag{3.21}$$

and

$$e(n) = d(n) - y(n). \tag{3.22}$$

Finally, the weight update equation is described by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\, e(n)\, \Lambda^{-2}\, \mathbf{T}\,\mathbf{u}(n). \tag{3.23}$$

The inverse of matrix Λ^2 accounts for the power normalization step of the algorithm. Since every filter coefficient is adapted independently there exists a corresponding normalization factor on the diagonal matrix

$$\Lambda^2 = \mathrm{diag}\begin{bmatrix} \sigma_0^2 & \sigma_1^2 & \ldots & \sigma_{M-1}^2 \end{bmatrix}^T \tag{3.24}$$

where

$$\sigma_i^2 = E\Big\{|x_i(n)|^2\Big\}. \tag{3.25}$$

The power of every independent component is usually approximated by an exponential weighted average [49,54] of past samples

$$\sigma_i^2(n) = \alpha\sigma_i^2(n-1) + |x_i(n)|^2, \qquad 0 < \alpha < 1. \tag{3.26}$$

Similar to the notion of NLMS in section 3.3, the energy might be insufficient in a time-varying signal like speech and music. This leads to ill-conditioning of the technique [54]. Since it is not trivial to predict the inactive periods of audio signals and pause the adaptation, a small positive constant is added to each of the diagonal elements of Λ^2

$$\widehat{\Lambda}^2 = \Lambda^2 + \epsilon \mathbf{I}. \tag{3.27}$$

Next the convergence properties of TDLMS are elaborated followed by a geometrical interpretation of the algorithm which will give a better understanding of the algorithm's principles.

3.4.3 Convergence properties of the TDLMS algorithm

Recalling that the LMS algorithm minimizes the MSE, it is useful to take a geometrical approach to analyze the convergence behavior of the adaptive filter. Thus, in this section more insight is obtained starting with following expression of the MSE surface

$$\xi(\mathbf{z}) = E\Big\{|e(n)|^2\Big\} = \xi_{\min} + \mathbf{z}^T \mathbf{R_u} \mathbf{z}, \tag{3.28}$$

where $\xi_{\min}$ is the mean square error evaluated at the Wiener solution $\mathbf{w}_{\text{opt}}$ and $\mathbf{z} = \mathbf{w} - \mathbf{w}_{\text{opt}}$. From Eq. (3.19) it can be shown that $\mathbf{R_x} = \mathbf{T}\,\mathbf{R_u}\mathbf{T}^T$. Hence, we can write

$$\xi(\mathbf{z}) = E\Big\{|e(n)|^2\Big\} = \xi_{\min} + \mathbf{z}^T \Big[\mathbf{T}\,\mathbf{R_u}\mathbf{T}^T\Big]\mathbf{z}. \tag{3.29}$$

At this point, it is of significant importance to state that the application of a unitary matrix $\mathbf{T}$ to the input vector is not sufficient to uncorrelate the signal. The input vector's transformation translates to a rotation of the contour lines of the MSE. Therefore, since the error surface is not modified there is no change in the convergence rate of the adaptation. This argument will be better established in a simple example further on in this section.

The power normalization of each independent component is carried out on the next stage, which in fact has the same effect as pre-whitening the input signal. In the case of a colored input signal, this is reflected to the conversion of the geometry of the error surface from hyperellipsoids into hyperspheres. Of course, when the filter's input is white Gaussian noise no improvement on the convergence speed of the algorithm is expected since all frequency bins have equal power. That means that all the points along each contour line of the MSE surface are equidistant from the optimal solution. By letting $\mathbf{z} = \Lambda^{-1}\tilde{\mathbf{z}}$ the error surface after the power normalization step of TDLMS is described by

$$\xi(\tilde{\mathbf{z}}) = E\Big\{|e(n)|^2\Big\} = \xi_{\min} + \tilde{\mathbf{z}}^T \Big[\Lambda^{-1}\mathbf{T}\,\mathbf{R_u}\mathbf{T}^T\Lambda^{-1}\Big]\tilde{\mathbf{z}}. \tag{3.30}$$

3.4.3.1 Geometrical Interpretation

A geometrical respresentation of the described algorithm is given next for a simple two-tap filter example. Let the autocorrelation a matrix of a colored input signal be the following

$$\mathbf{R_u} = \begin{bmatrix} 1.0 & 0.9 \\ 0.9 & 1.0 \end{bmatrix}. \tag{3.31}$$

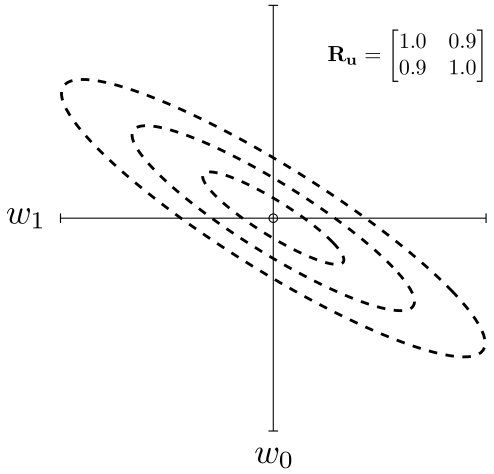
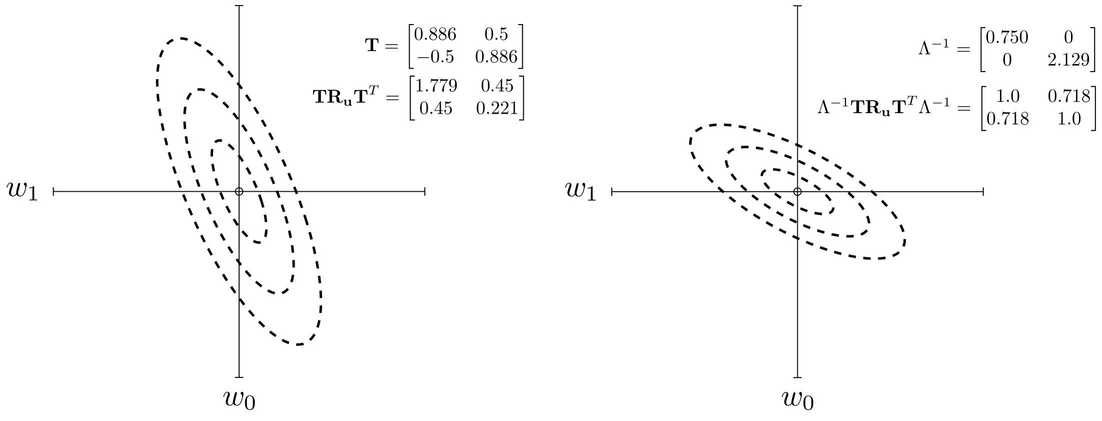


Figure 3.6: The contour lines of the MSE surface.

The 3d-surface of the mean square error described by Eq. 3.28 is depicted in Fig. 3.6. One of the fundamental assumptions in adaptive filtering is that $\mathbf{R_u}$ is a positive definite matrix which is the reason why the contour lines have this hyperelliptic shape. The length of the hyperellipses' semiaxes are in fact proportional to the eigenvalues of $\mathbf{R_u}$. Hence in this case, the bigger the ratio between the maximum and the minimum length, the slower the convergence of the adaptive filter is expected. Applying a unitary transformation matrix $\mathbf{T}$ to the input would align the error surface with the principle coordinates of the system. The result of this transformation with an arbitrary matrix is depicted in Fig. 3.7a. Finally, the last step of the method is performed which consists of the power normalization of the transformed signal. The effect of this operation is shown in Fig. 3.7b.

Note that the contour lines are not completely spherical as suggested earlier in this section and this traces back to the suboptimal transformation matrix that has been used for this example. Since $\mathbf{T}$ is fixed and data independent, it fails from making the autocorrelation matrix diagonal and thus the signal is still partially correlated. This is also the case when using fixed transforms like the DFT and the DCT. Most of these transforms are represented by non-ideal band-pass filters which results in spectral leakage [49].



(a)

(b)

Figure 3.7: The contour lines of the MSE surface after orthogonalizations and normalization. Note that the (a) orthogonalization step aims at the alignment of the hyperellipses with the coordinate axes, whereas (b) power normalization achieves partial decorrelation of the input signal by making the contours more spherical.

3.5 Simulations

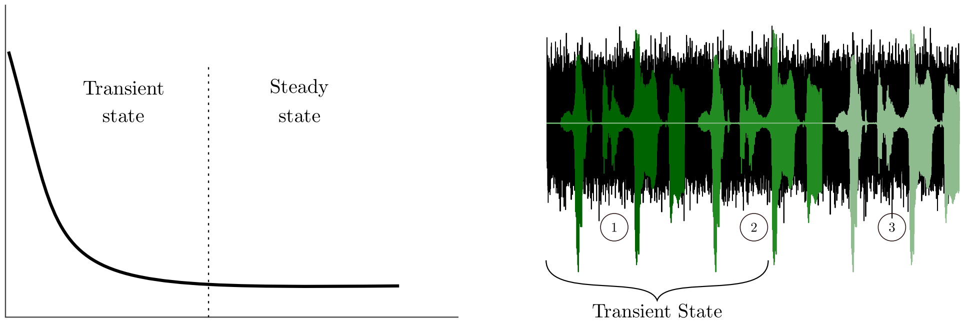
The performance of the described algorithms are assessed in a simulated environment considering an ANC scenario. The noise signal is convolved with an arbitrarily initialized 50-tap IR and is mixed subsequently with a 6 sec speech signal sampled at 16 kHz. The adaptive filter and the unknown system are assumed to have the same length. The SNR of the noisy speech signal is set at 0 dB. Since all of the investigated algorithms approximate the Wiener solution $\mathbf{w}_o$ in the steady state, the convergence speed of the filter's adaptation will be the main performance criterion. More specifically, the mean squared deviation (MSD) of the adaptive filter is plotted with respect to every iteration. This metric of MSD measures the average difference between the adaptive filter and the optimal solution, i.e. $E\Big\{\|\mathbf{w}(n) - \mathbf{w}_o\|^2\Big\}$ [38].

However, the MSD cannot be measured in practice and therefore as $\mathbf{w}_o$ the initialized 50-tap IR is considered in this framework and the MSD is calculated as the average $\frac{1}{M}\sum_{i=1}^{M}\Big(w_i(n) - w_{o,i}\Big)^2$, where $w_i(n)$ is the i-th coefficient at iteration n and M is the number of the filter coefficients, in this case $M = 50$.

Ultimately, the output SNR is evaluated for both investigated scenarios for a set of very low SNRs. Recalling from section 3.1.1, due to the orthogonality principle the error signal of the adaptive filter (see Fig. 3.2) gives an approximation of the speech signal. Therefore the output SNR is computed as follows

$$\mathrm{SNR}_o = 10\log_{10}\frac{\sum_{n=k}^{L} s^2(n)}{\sum_{n=k}^{L}\Big(s(n)-e(n)\Big)^2}, \tag{3.32}$$

where $L-k$ is the time interval where speech is present. Nonetheless, the error signal is formed during the filter's adaptation which implies that in case of slow convergence the filter might not have reached the steady mode of operation. In Fig. 3.8a the transient and steady state regions of the filter's learning curve are demonstrated. In a real ANC scenario used for command recognition the adaptation of the filter is expected to have started much sooner than the speech segment is incorporated in the system. Obviously, the output SNR is highly dependent on the position of the speech signal within noise. Therefore, in the simulations in order to avoid the early transient state of the filter's adaptation, the speech segment is mixed with the late part of the noise. An example of a speech segment mixed with different regions of noise is illustrated in Fig. 3.8b.

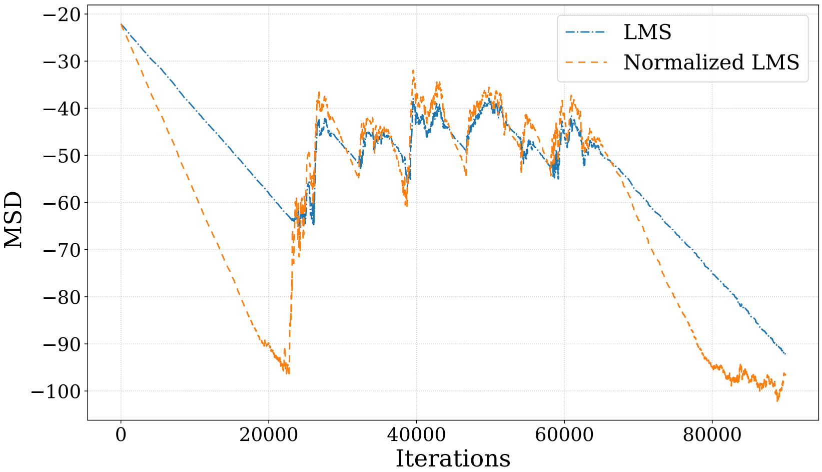


(a) Operation regions of the adaptive filter learning curve.

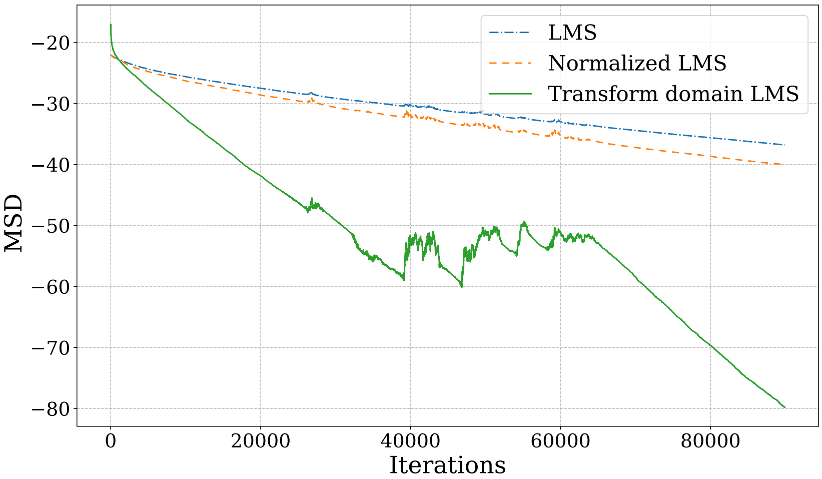
(b) Speech segment mixed with three different parts of the noise signal.

Figure 3.8: Speech and music mixing configuration. The steepness and length of the transient state shown in (a) are the main characteristics of the adaptive filter's convergence. To avoid the early transient phase of the filter speech is assumed in positions 2 or 3 as depicted in (b).

In the first experiment, the LMS and NLMS algorithms are compared considering white Gaussian noise in the adaptive filter's input with variance $\sigma_u^2 = 0.01$. The filter coefficients for LMS and NLMS are updated according to Eq. (3.12) and (3.16) respectively. For comparison purposes the step size of both LMS and NLMS algorithms has been set to $\mu = 0.01$. In Fig. 3.9a the MSD of the compared methods is demonstrated where it is clear that NLMS outruns LMS in terms of convergence speed. Moreover, both algorithms when they operate in the steady state mode they have comparable performance, an observation that is expected since both algorithms approximate the Wiener solution.



(a) Convergence of LMS and NLMS for white Gaussian noise. Note that for the same fixed step size $\mu = 0.01$ NLMS converges faster than LMS.



(b) Convergence of LMS and NLMS for brownian noise (random walk noise). Note the significant improvement in the convergence speed of TDLMS compared to the other two methods.

Figure 3.9: Mean squared deviation (MSD) for both investigated experiments. The MSD measures the average difference between the adaptive filter $\mathbf{w}(n)$ and the optimal Wiener solution $\mathbf{w}_o$, i.e. $E\{\|\mathbf{w}(n) - \mathbf{w}_o\|^2\}$.

In the second experiment, the case of having colored noise in the input of the adaptive filter is investigated. More specifically, Brownian noise (red noise) is generated, described by the first-order autoregressive process

$$u(n) = 0.85\,u(n-1) + \delta(n),$$

where $\delta(n)$ is white Gaussian noise of variance $\sigma_\delta^2 = 0.16$. Here, in addition to LMS and NLMS, the algorithm of TDLMS is tested. The filter in the latter method adjusts its coefficients according to Eq. (3.23). Moreover, a DCT matrix is applied to the input signal $u(n)$ given by

$$\mathbf{T}_{\text{DCT}}(i,j) = \sqrt{\frac{2}{N}}\, K_i \cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right) i\right] \qquad i,j = 0,\ldots,M-1$$

with $K_i = 1 \,/\, \sqrt{2}$ for $i = 0$ and 1 otherwise [53]. Regarding the power normalization, which comprises the second stage of TDLMS, an estimate of the power of every component is considered using the exponential weighted average described in Eq. (3.23). The forgetting factor of this weighted average was set to $\alpha = 0.999$. The MSD of the compared algorithms for this experiment is demonstrated in Fig. 3.9b. It is clear that the TDLMS algorithm outperforms the rest of the algorithms in terms of convergence rate. The dependence of LMS convergence on the conditioning of the input autocorrelation matrix $\mathbf{R_u}$ was described earlier in Eq. (3.13). Furthermore, in Fig. 3.10 the eigenvalues of $\mathbf{R_u}$ are plotted before and after applying the DCT followed by the power normalization. The plot indicates the significant reduction in the eigenvalue spread $\kappa(\mathbf{R_u})$, which in fact here drops from 141 to 2. In addition, Fig. 3.11 depicts the autocorrelation matrix before and after the transformation of the input signal. The non-zero elements around the diagonal of $\mathbf{R_u}$, depicted in Fig. 3.11a, point out that Brownian noise is correlated. On the other hand, Fig. 3.11b confirms the efficiency of TDLMS in diagonalizing $\mathbf{R_u}$ and normalizing the power across the independent components of the input.

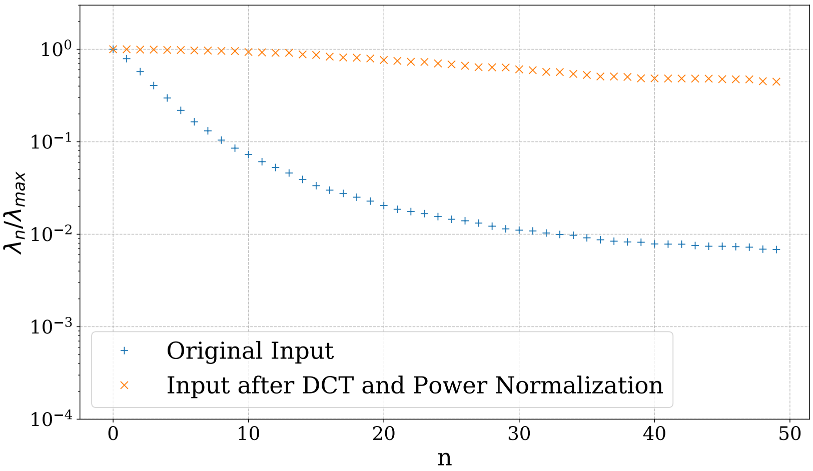
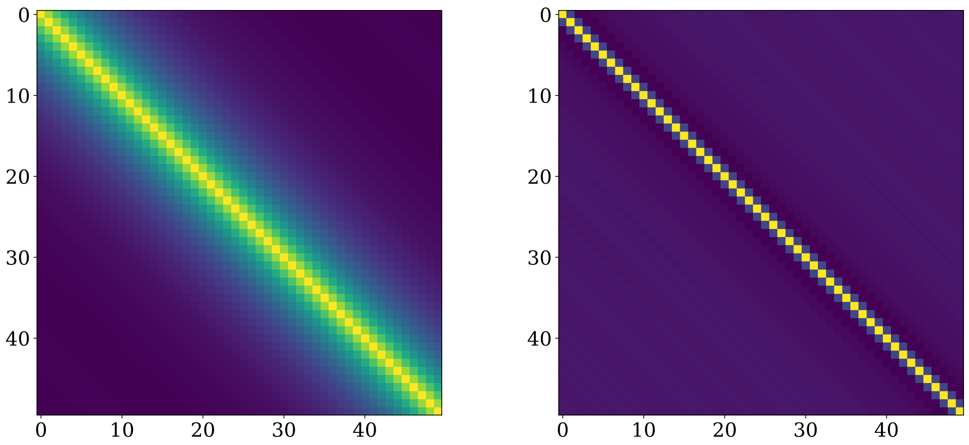


Figure 3.10: Eigenvalues of the the input autocorrelation matrix. Note that the ratio between the largest and the smallest eigenvalue has dropped significantly after the application of DCT and power normalization ($\kappa(\mathbf{R}) : 141 \rightarrow 2$).

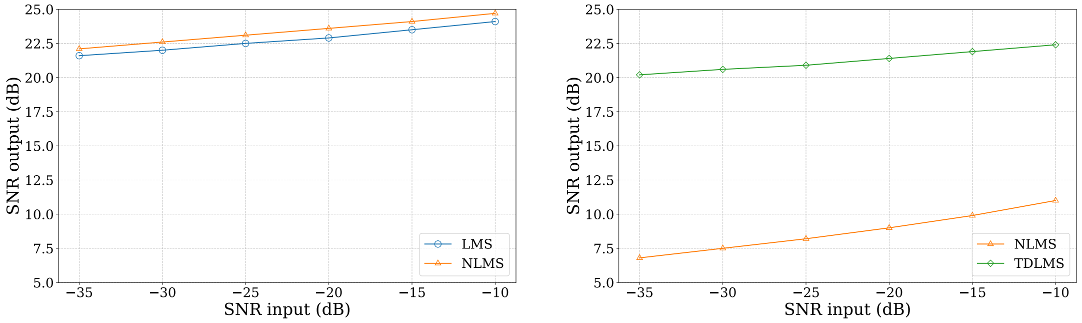


(a)

(b)

Figure 3.11: Autocorrelation matrix **R** of Brownian noise (a) before and (b) after the DCT and power normalization. The non-zero elements around the diagonal of **R** indicate correlation of the noise which is eliminated in (b).

Finally, in Fig. 3.12 the output SNR is plotted for a set of low input SNRs. The values of this set were chosen in correspondence to the expected input SNRs reported in the Table 2.1 of the previous chapter. For the case of white Gaussian noise (Fig. 3.12a) we observe that LMS and NLMS produce similar results since in both cases the filter reaches the steady state in comparable time instances. However, the correlated Brownian noise causes the NLMS adaptive filter to converge much slower justifying its degraded performance compared to the TDLMS digital filter (Fig. 3.12b).



(a) Comparison of LMS and NLMS for white Gaussian noise.

(b) Comparison of NLMS and TDLMS for Brownian noise.

Figure 3.12: Output SNR regarding both investigated experiments.

4 Results

In this chapter the results from a Monte Carlo simulation are demonstrated regarding the performance of the NLMS and TDLMS algorithms described in the previous chapter. In particular, the performance metrics that have been used to evaluate the aforementioned techniques are the output SNR of the ANC system, STOI as an objective measure of speech intelligibility and WER which was computed after feeding the enhanced signals to a machine learning based ASR algorithm.

4.1 Simulation set-up

The speech signals used in the Monte Carlo simulation are part of the GRID [57] database. More specifically, 24 speech segments were selected corresponding to 4 female and 4 male speakers. Each of these segments were mixed with 6 sec of music as shown in Fig. 4.1. The music database consists of 5 tracks of different genre normalized with respect to their loudness level. Both speech and music signals were downsampled to 16 kHz. The speech segment in every case is mixed deliberately with the late part of the music segment in order to avoid the transient state of the adaptive filter's convergence. For more details on the filter states the reader is referred to Section 3.5.

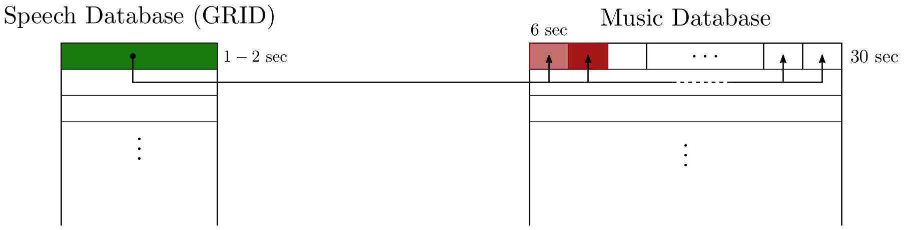


Figure 4.1: Monte Carlo simulation setup. 24 speech segments were mixed with music segments of different genre (classical, rock, jazz, pop).

The impulse responses (IRs) that the speech and the music signals have been convolved with, correspond to real measurements carried out in both an anechoic and a reverberant environment. The reverberant room's volume is $62\,\mathrm{m}^3$ and has a reverberation time $T_{60} = 0.3\,\mathrm{sec}$. In the case of the reverberant environment the speech source was considered at five different distances d_s with respect to the music source, namely 150, 200, 300, 400 and 500 cm. On the other hand, due to lack of space in the anechoic environment, it was possible to measure IRs considering the talker up to $d_s = 300\,\mathrm{cm}$ away from the loudspeaker. In both environments the loudspeaker microphone was placed at a distance of $d_m = 20\,\mathrm{cm}$ from the music source.

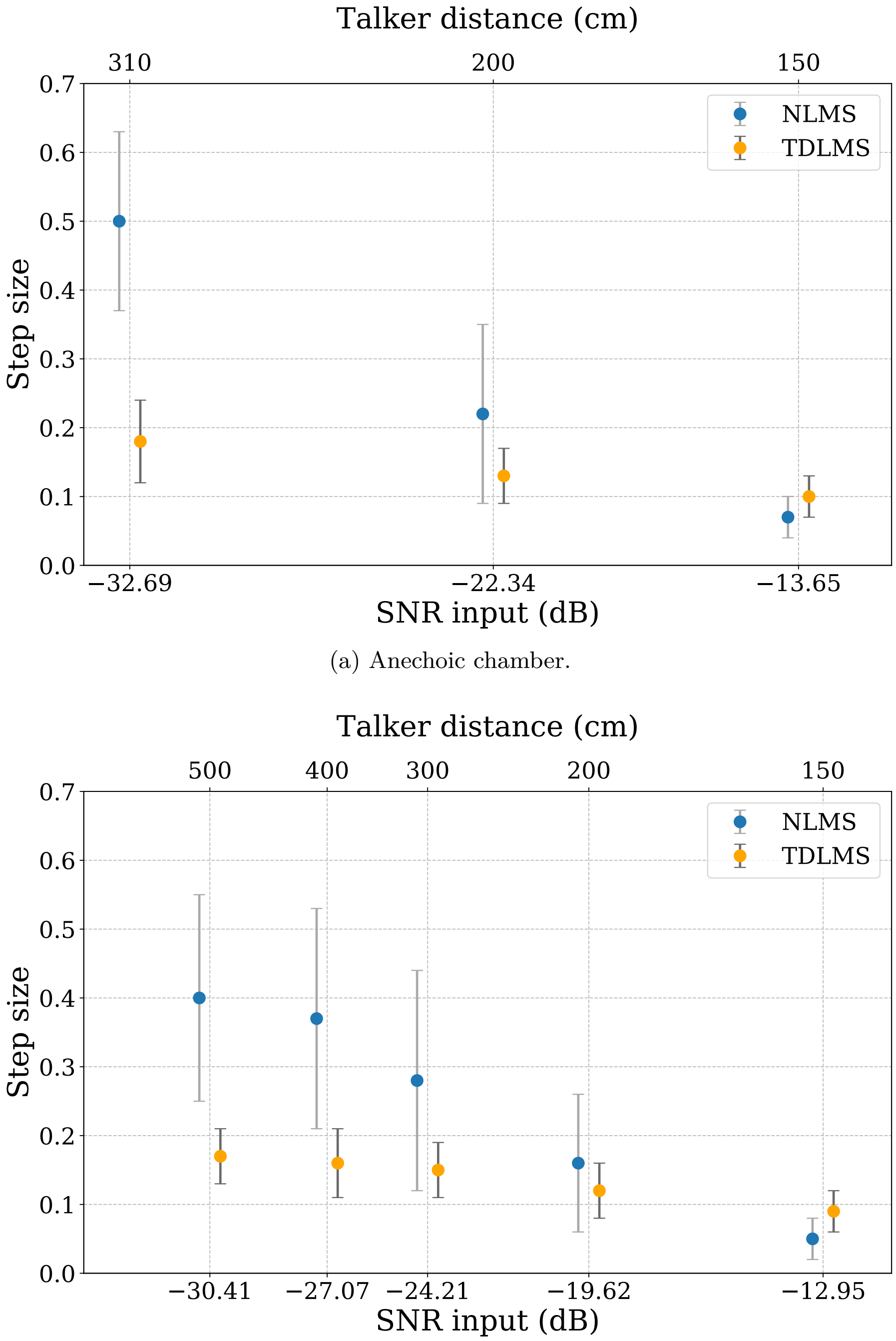
The setup for recording individual IRs was the following. The music and the speech source were represented by two loudspeakers. An additional microphone placed at 1 m in front of the speech source captured individual sweeps generated by each source. For every IR, 4 sweeps[2] were generated and were subsequently averaged. The pressure level of each sweep was considered at $70\,\mathrm{dB}_{\mathrm{SPL}}$ on average (see Fig. 2.1) across the frequency range.

The IRs were recorded at a sampling rate of 48 kHz and were subsequently resampled at 16 kHz. They were then truncated to the noise floor of the IR measured in the reverberant room considering the speech source at $d_s = 150\,\mathrm{cm}$.

The algorithms were implemented and evaluated in a Python 3.6 environment. The fundamental parameters of the adaptive filters were determined as follows: The length M of the filter was set to 500. The regularization parameter that accounts for numerical issues in the normalization factor of both NLMS and TDLMS was set to $\epsilon = 0.001$. The forgetting factor of the exponential weighted average in TDLMS was chosen to be $\alpha = 0.999$. Being that the step size controls the learning rate of the adaptive filter in both algorithms, its value has a great impact on the final results. However, optimizing for its value is out of the scope of this work. Therefore, the value that yields the maximum output SNR was chosen as the step size in this evaluation. This was accomplished by performing a linear search in every iteration of the Monte Carlo simulation.

In Fig. 4.2 the step size of both algorithms is plotted. It can be observed that as the SNR drops, both the step size of NLMS and TDLMS increases. However, we see that in NLMS the step size varies significantly as a function of the input SNR. In addition, by taking a look at the standard deviation, it can be seen that the step size of TDLMS lies in a much smaller interval compared to NLMS for each set of Monte Carlo simulation. This indicates that TDLMS offers a better control of the step size parameter that plays a significant role in the filter's convergence.

[2] As sweep, a sinusoidal signal that sweeps usually through the entire frequency range of 20 - 20,000Hz is considered.



(a) Anechoic chamber.

(b) Reverberant room.

Figure 4.2: The mean and standard deviation of the step size for NLMS and TDLMS with respect to the input SNR. Note that TDLMS has a significantly smaller standard deviation suggesting more consistent results.

4.2 Performance metrics

In this section the performance metrics considered for the proposed algorithms are defined. Note that all metrics are plotted over the mean values of the input SNRs which correspond to different talker positions as denoted on the top axis of every graph. Let the signal captured by the microphone be $d(n) = s(n) + z(n)$ where $s(n)$ is the clean speech signal and $z(n)$ is the music signal. Then, the input SNR is defined as

$$\mathrm{SNR}_i = 10\log_{10}\frac{\sum_{n=k}^{L} s^2(n)}{\sum_{n=k}^{L} z^2(n)}. \tag{4.1}$$

In the above equation, it is assumed that $L-k$ is the discrete time interval within $d(n)$ where speech is present.

4.2.1 Signal-to-Noise Ratio (SNR)

The first performance metric that was computed was the output SNR of the ANC system and is plotted with respect to the input SNR at the microphone. Moreover, since the residual noise in the enhanced signal cannot be measured, the output SNR is calculated as follows

$$\mathrm{SNR}_o = 10\log_{10}\frac{\sum_{n=k}^{L} s^2(n)}{\sum_{n=k}^{L}\Big(s(n)-\hat{s}(n)\Big)^2}, \tag{4.2}$$

where $\hat{s}(n)$ is the estimated speech signal that coincides with the error signal of the adaptive filter (see Fig. 3.2) as proved in Section 3.1.1.

4.2.2 Short Time Objective Intelligibility (STOI)

As an objective speech intelligibility measurement, STOI was used. It is important to state at this point that STOI does not follow a linear relationship with respect to speech intelligibility. In other words, the improvement in the intelligibility of speech in the interval $[0.6, 0.8]$ is significantly larger compared to the interval $[0.8, 1.0]$. More insight about the correlation between subjective speech intelligbility and the metric of STOI can be found in [58].

4.2.3 Word Error Rate (WER)

Ultimately, since the proposed framework aims at supporting the front end of a speech recogniser, the word error rate (WER) is calculated

$$\mathrm{WER} = \frac{S+D+I}{N}\,\%, \tag{4.3}$$

where S is the number of substitutions, D the number of deletions, and I, the number of insertions in the hypothesis sentence corresponding to the extracted speech. N is the number of the words in the reference sentence. Both the hypothesis and reference sentences were obtained through the speech-to-text engine, DeepSpeech. A deep learning approach was used in the development of this open source ASR implementation.

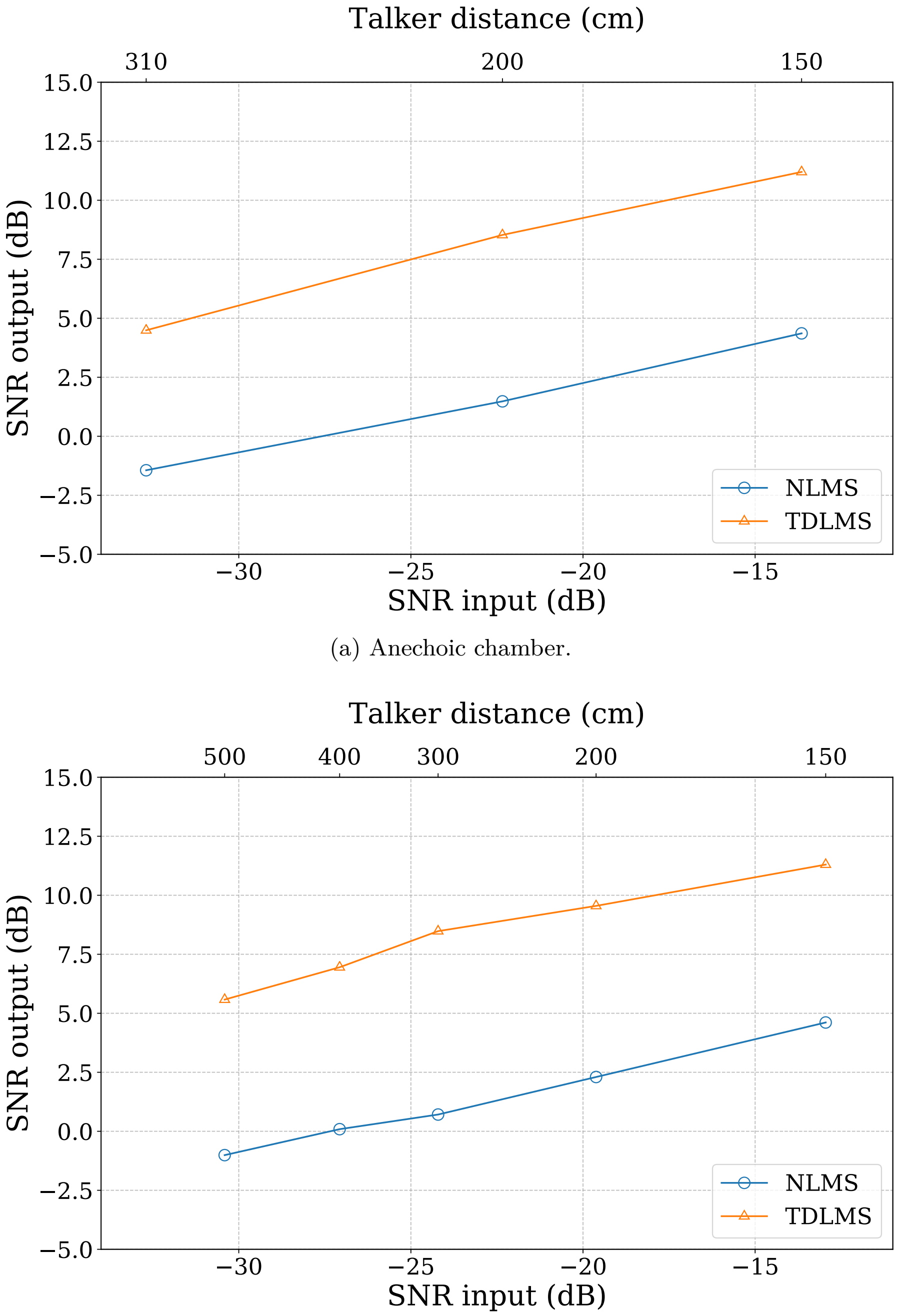
4.3 Monte Carlo results

In this section, the results of the Monte Carlo simulation are presented. The performance of NLMS and TDLMS is assessed regarding their effectiveness in suppressing the interfering music signal in significantly low input SNRs. In addition, the performance of the two methods is compared for all three performance metrics introduced in the previous section. Each performance metric is illustrated as a function of the input SNR. As the top axis of every graph indicates, the values of the input SNR correspond to different distances of the speech source from the music source. In addition, the plots below show the results of each metric in both the anechoic chamber and the reverberant room.

In Fig. 4.3 the output SNR is illustrated for both NLMS and TDLMS. It is obvious that both NLMS and TDLMS succeed in suppressing significantly the music signal that corrupts speech. In general, as the input SNR increases the output SNR also increases in a linear manner. It can also be observed that the output SNR is comparable in both the anechoic and the reverberant room when considering the talker at distances close to the microphone. Specifically, for the talker at a distance of 150 cm from the music source, the output gain is approximatively 24 dB in both environments. However, it can be seen that as the talker moves away from the noise source, the input SNR drops considerably faster in the anechoic chamber in comparison to the reverberant room and consequently so does the output SNR. Further details on the effect of reverberation in the input SNR are given in Section 4.4.

The error bars in Fig. 4.4 depict the the mean and standard deviation of the metric STOI regarding the intelligibility of the enhanced signals. Overall, the results of TDLMS for both the anechoic and the reverberant environment indicate that the enhanced speech is intelligible even at very low input SNRs. Moreover, it can be observed that the results in NLMS show a larger standard deviation than TDLMS. This implies that more consistent results are obtained using the latter method.

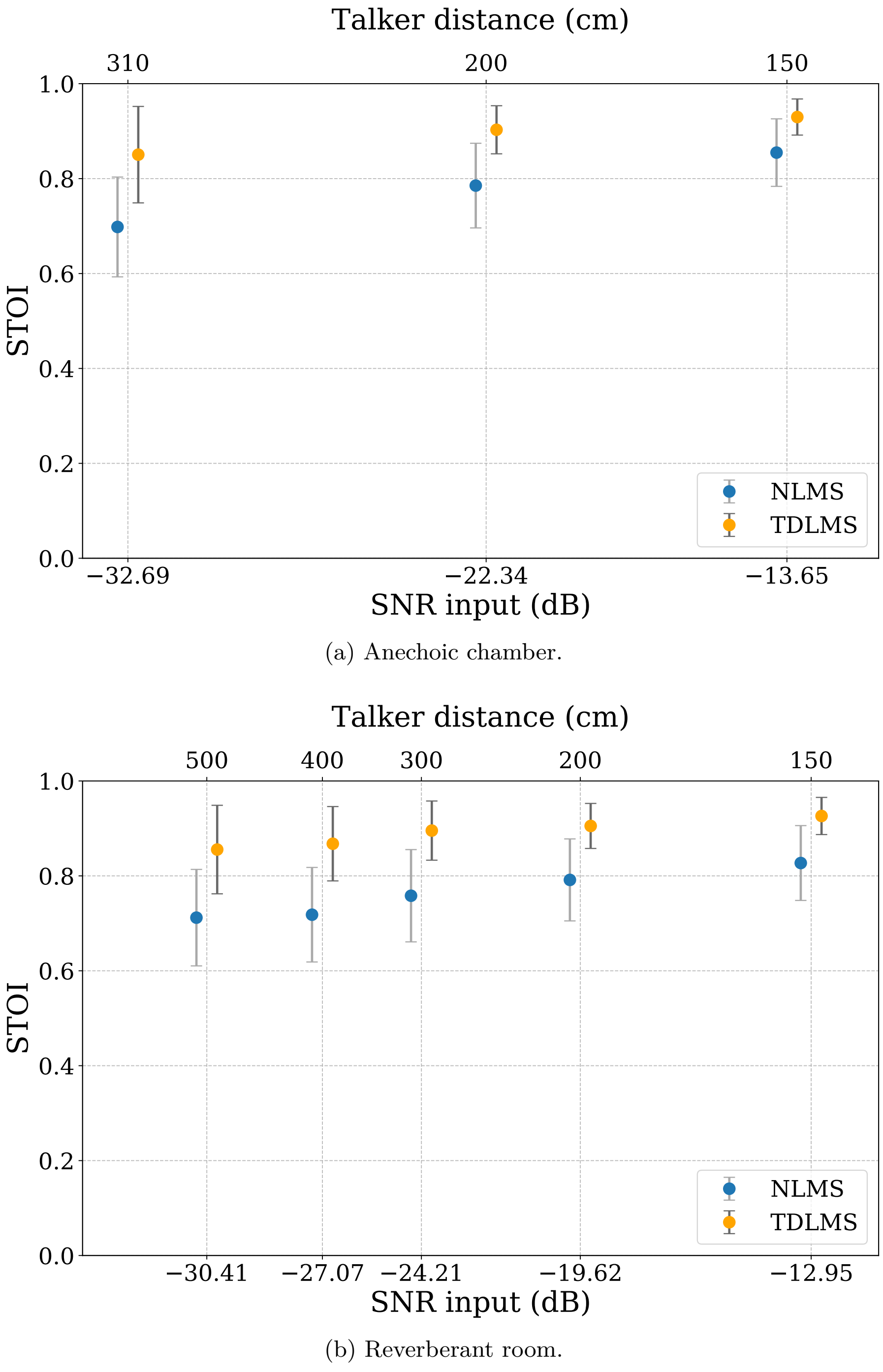
Finally, in Fig. 4.5 the results on the WER are depicted as function of the input SNRs which again correspond to the different talker positions in each environment. In general, it is seen that for the whole range of input SNRs investigated, the WER is 100 % which implies that the speech recognizer fails completely in predicting



(a) Anechoic chamber.

(b) Reverberant room.

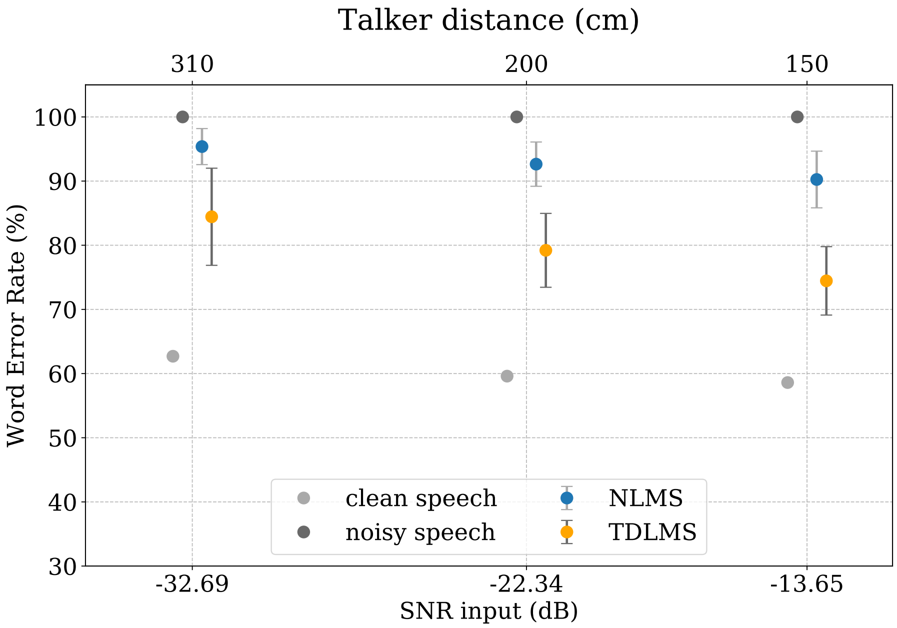
Figure 4.3: Output SNR for (a) the anechoic chamber and (b) the reverberant room. Note the large gain in the SNR for both NLMS and TDLMS. TDLMS clearly outperforms NLMS due to its faster convergence.



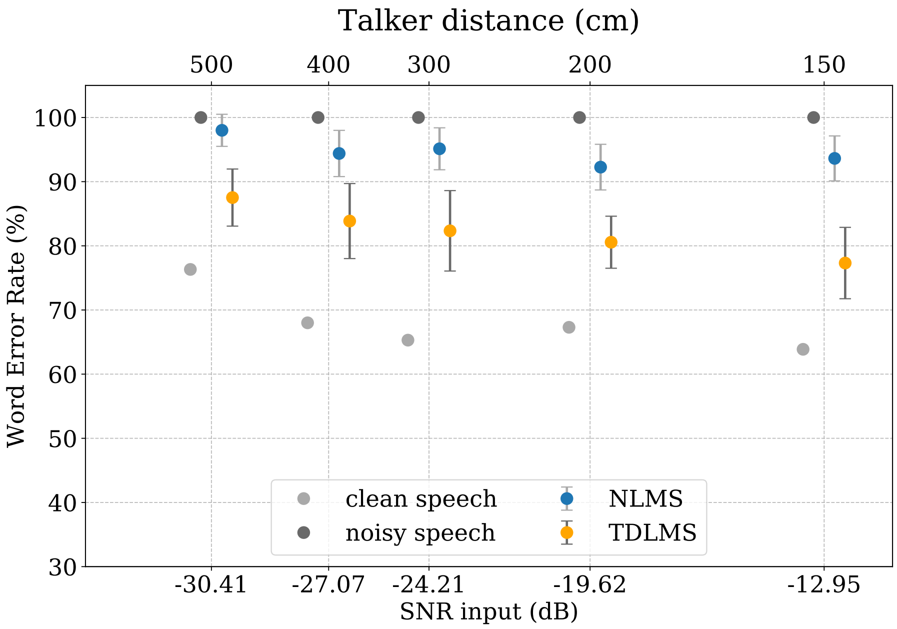
(a) Anechoic chamber.

(b) Reverberant room.

Figure 4.4: The mean and standard deviation of STOI for (a) the anechoic chamber and (b) the reverberant room. STOI measures objective intellibility on a scale from 0 to 1. Note that even for very low SNRs, TDLMS attains on average high STOI scores in both tested environments.



(a) Aechoic chamber.



(b) Reverberant room.

Figure 4.5: The mean and standard deviation of WER for (a) the anechoic chamber and (b) the reverberant room. The mean and standard deviation of WER regarding clean speech after being convolved with the corresponding IR (light grey), noisy speech as it is captured by the microphone (dark grey), the enhanced speech using NLMS (blue) and the enhanced speech using TDLMS (orange).

the words in the reference sentence. As a general remark, it should be noted that the overall performance of both speech enhancement algorithms is poor. In addition, it is also observed that the WER results of the clean speech are not satisfactory either, as one would expect (on average 60 % for the anechoic chamber and 70 % for the reverberant room). This suggests that in general the results of the calculated WER are heavily dependent on the speech recognizer used. More insight on this observation is given in the discussion section that follows. Nonetheless, the performance of TDLMS is considerably higher to that of NLMS.

4.4 Discussion

This section focuses on pointing out the findings and learnings of this work. Moreover, the performance of both NLMS and TDLMS is summarized for both the anechoic chamber and the reverberant room in the Tables 4.1 and 4.2 respectively.

First of all, it is observed that the values regarding the prior investigation on the expected input SNRs shown in Table 2.1, seem to agree with the input SNRs resulting from the measurements in the anechoic chamber. On the other hand, in the reverberant room, it can be seen that the SNR is dropping at a slower rate with increasing distance of the talker. That occurs mainly due to the reverberation which is absent in the anechoic chamber. In other words, the energy decay in the reverberant room is longer compared to an anechoic environment. In particular, the reverberant tail from the speech source IR has a great contribution to the energy of the corresponding signal. Therefore higher SNRs are attained in the reverberant room compared to the anechoic chamber.

By inspecting the output SNRs of both the NLMS and TDLMS algorithms the gain that is achieved is considerably large and as the input SNR decreases it gets larger. This due to the fact that the adaptive filter in an ANC scenario aims at finding the best linear mapping of its output and the observed signal. In very low input SNRs the music component is quite dominant compared to the speech signal. As a consequence, the adaptive filter manages to produce an output significantly close to the music profile of the microphone signal. The better the estimation of the music component the larger is its suppression leading to high output SNRs.

Furthermore, the STOI results seem to present the same trend as the output SNR. In particular, as the output SNR inreases so does the score of STOI. Of great interest, however, is the fact that even for extremely low SNRs, namely $-30\,\mathrm{dB}$, the enhanced signals using TDLMS have a STOI score above 0.8. It is also observed that the standard deviation of both methods grows as the input SNR drops. This suggests that the precision of the results increases when moving towards higher SNRs.

		NLMS			TDLMS		
d_s	SNR$_i$	SNR$_o$	STOI	WER	SNR$_o$	STOI	WER
150 cm	-13.5	4.4	0.85	90.2%	11.2	0.93	74.5%
200 cm	-22.3	1.5	0.79	92.6%	8.5	0.90	79.2%
300 cm	-32.7	-1.4	0.70	95.4%	4.5	0.85	84.4%

Table 4.1: Overall comparison of NLMS and TDLMS regarding the simulation in the anechoic chamber. The microphone loudspeaker is considered at distance $d_m = 0.2\,\mathrm{m}$ from the music source and the talker at d_s.

		NLMS			TDLMS		
d_s	SNR$_i$	SNR$_o$	STOI	WER	SNR$_o$	STOI	WER
150 cm	-13.0	4.6	0.83	93.6%	11.3	0.93	77.3%
200 cm	-19.6	2.3	0.79	92.3%	9.5	0.91	80.6%
300 cm	-24.2	0.7	0.76	95.1%	8.5	0.90	82.3%
400 cm	-27.1	0.1	0.72	94.4 %	7.0	0.87	83.9%
500 cm	-30.4	-1.0	0.71	98.4%	5.6	0.86	87.5%

Table 4.2: Overall comparison of NLMS and TDLMS regarding the simulation in the reverberant room. The microphone loudspeaker is considered at distance $d_m = 0.2\,\mathrm{m}$ from the music source $d_m = 0.2\,\mathrm{m}$ and the talker at d_s.

Ultimately, the WER results seem rather discouraging at first for both tested adaptive schemes. Since the WER even for the clean speech signals were unsatisfactory a closer look was taken on the transcripted raw speech segments of the database. It was observed that the implementation of the ASR DeepSpeech tends to make meaningful sentences out of the input sequences. It can be therefore concluded, being that the GRID database consists of speech segments with certain structure but meaningless content, the ASR system fails to produce desirable results. Moreover, the fact that DeepSpeech is a context-based ASR implementation is justified by the results demonstrated in Table 4.3.

	example 1	**example 2**
reference :	lay white in P five now	lay blue with U five please
clean speech :	a white in tea five now	lably with you five please
enhanced speech (NLMS) :	palywye five	-
enhanced speech (TDLMS) :	lay one in a pet five now	i mean with you find

Table 4.3: DeepSpeech trasnscription results.

5 Conclusions

5.1 Outlook

Speech enhancement is a signal processing technique that has been addressed from many different perspectives as seen in the literature. The problem that was posed in this thesis is the isolation of a voice command corrupted with a music signal played by a smart loudspeaker. The incentive behind this notion is to support the speech recognition system of the loudspeaker with a speech enhancement framework. In other words, this works aims at achieving low WERs at the output of the ASR system without having to reduce the volume of the music played when the user interacts with the smart speaker. Undoubtedly, the main challenge of this work is the suppression of the interfering music signal considering significantly low SNRs.

One of the primary investigations of this work was to determine the expected SNRs in the loudspeaker microphone. Thus, a ball-park scenario was examined where the talker and the loudspeaker sensor where considered in different positions in the free field. The results of this investigation came to an agreement with an experiment conducted in an anechoic chamber. In contrast, the experiment in the reverberant room resulted in higher SNRs owing at a great extent to the reverberant component of the speech signal.

Eventually, the concept of adaptive filtering seemed to fit best this study and more explicitly the scheme of ANC. In prinicple, ANC is a technique for cancelling the noise within an observation by solving the least squares problem iteratively. The methods that were studied and evaluated belong to the LMS family where the MSE is minimized. Such algorithms have proved to be favorable candidates for real time application due to their low computational complexity and robustness against quantization errors.

Among other methods, the traditional LMS, the normalized LMS (NLMS) and the transform domain LMS (TDLMS) were studied. The traditional LMS and NLMS, being derived from the gradient descent and the Newton method respectively, suffer from slow convergence when the input signal is highly colored. This problem is tackled by TDLMS by introducing two stages to the traditional LMS update scheme. Specifically, the input signal undergoes at first an orthogonal transformation which is followed by the normalization of every transformed component. In practice, as shown in Section 3.4 these two steps aim to make the contour lines of the MSE surface circular which is equivalent to whitening the input signal. Major role on the performance of TDLMS plays the orthogonal transform used. KLT is the optimal transform in terms of energy compaction. Since though KLT is data dependent and thus computationally expensive, for a real time application, fixed transforms are preferred instead. Amongst other linear transformations DCT has shown to have the closest to the KLT decorrelation properties and therefore it was considered in the evaluation of this framework.

To assess the effectiveness of the adaptive filtering algorithms in suppressing colored noise within a corrupted speech signal, a set of Monte Carlo simulations was conducted. Real IRs measurements were made both in an anechoic chamber and in a revereberant room. Although both NLMS and TDLMS managed to suppress significantly the music signal the latter technique produced superior and more consistent results especially in extremely low SNRs as demonstrated in Section 4.3.

Overall, it can be concluded that providing the interfering signal is available, adaptive filtering proved to be particularly effective in suppressing the music that corrupts the speech signal. Finally, due to its convergence properties, TDLMS outperforms NLMS as all of the evaluated metrics have shown.

5.2 Future Work

To conclude, the work of this thesis can be considered the beginning of further investigation in the field of colored noise cancellation. The following research directions are therefore proposed.

Since the recommended speech enhancement framework comprises a pre-processing stage in the front-end of a speech recogniser a benchmark on different ASR systems would be more indicative on the performance of TDLMS in terms of WER. Undoubtedly, it was also observed that the absent structure of the sentences used in the Mont Carlo simulation had a great effect on the WER. Therefore, it is expected that using a database with structured speech and syntax would produce improved results. x

In the investigated scenario, as noise only the music signal played from the loudspeaker is considered. That implies that the output SNR in the proposed framework is bounded by the potential environmental noise that might exist. Hence, it is suggested that noise reduction from different sources is also considered in the framework either as prior information in the reference input of the adaptive filter or as a post processing technique. Undoubtedly, it should be highlighted that in any case, the computational complexity should be taken into account for that the framework should operate in real time.

Secondly, more loudspeakers can be considered in the current setup which entails that spatial information can be incorporated in the system. Providing that the relative positions of the music sources are also known, beamforming can be used leading in more effective noise reduction.

The convergence of adaptive algorithms depends on a number of factors. The parameter that controls how fast the adaptive scheme reaches the solution is the step size. In this work, a fixed step size that corresponds to the maximum SNR at the output of the ANC system is used. Since the value of the step size can vary considerably, incorporating a time-varying step size like the ones derived in [59,60] would assist the adaptive algorithm in achieving even faster convergence rates.

Finally, the unitary transform in TDLMS plays a crucial role in decorrelating the input signal. It has been observed that the use of DCT results in a significant reduction in the condition number of the input autocorrelation matrix. Thus, an adaptive signal-dependent unitary transform may present a better performance although at the expense of higher computational requirements.

A Acronyms

SNR	signal-to-noise ratio
WER	word error rate
STOI	short time objective intelligibility
ASR	automatic speech recognition
SPL	sound pressure level
ANC	adaptive noise cancellation
ATF	acoustic transfer function
RTF	relative transfer function
WSS	wide-sense stationary
BSS	blind source separation
ICA	independent component analysis
LMS	least mean squares
NLMS	normalized least mean squares
TDAF	transform domain adaptive filters
TDLMS	transform domain least mean squares
RLS	recursive least squares
DCT	discrete cosine transform
KLT	Karhunen Loéve transform

WHT	Walsh-Hadamard transform
PO2	powers-of-2 transform
DHT	discrete Hartley transform
FIR	finite impulse response
TDL	tapped delay line
IIR	infinite impulse response
MSE	mean squared error
MSD	mean squared deviation
DSB	delay and sum beamformer
MVDR	minimum variance distortionless response
LCMV	linear contraint minimum variance
GSC	generalized sidelobe canceler
DFT	discrete Fourier transform
MVDR	minimum variance distortionless response

Bibliography

[1] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 745–777, 2014.

[2] F. Jacobsen and P. M. Juhl, *Fundamentals of general linear acoustics.* John Wiley & Sons, 2013.

[3] J. J. Kellaris, S. P. Mantel, and M. B. Altsech, "Decibels, disposition, and duration: the impact of musical loudness and internal states on time perceptions," *ACR North American Advances*, 1996.

[4] Engineering Toolbox, "Voice level at distance," 2005.

[5] Industrial Noise Control, "Comparitive examples of noise levels," 2018.

[6] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 229–238, 2008.

[7] J. Ma, Y. Hu, and P. C. Loizou, "Objective measures for predicting speech intelligibility in noisy conditions based on new band-importance functions," *The Journal of the Acoustical Society of America*, vol. 125, no. 5, pp. 3387–3405, 2009.

[8] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[9] A. Moore, P. P. Parada, and P. Naylor, "Speech enhancement for robust automatic speech recognition: Evaluation using a baseline system and instrumental measures," *Computer Speech & Language*, vol. 46, pp. 574–584, 2017.

[10] Mozilla Machine Learning Group, "DeepSpeech, a tensorflow implementation of baidu's deepspeech architecture," 2014.

[11] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[12] M. Brandstein and D. Ward, *Microphone arrays: signal processing techniques and applications*. Springer Science & Business Media, 2013.

[13] J. Benesty, J. Chen, Y. Huang, and J. Dmochowski, "On microphone-array beamforming from a mimo acoustic signal processing perspective," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1053–1065, 2007.

[14] A.-J. van der Veen and G. Leus, "Signal processing for communications," *Delft University of Technology*, p. 109, 2005.

[15] I. Cohen, J. Benesty, and S. Gannot, *Speech processing in modern communication: Challenges and perspectives*, vol. 3. Springer Science & Business Media, 2009.

[16] R. A. Monzingo and T. W. Miller, *Introduction to adaptive arrays*. Scitech publishing, 2004.

[17] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New insights into the noise reduction wiener filter," *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1218–1234, 2006.

[18] K. U. Simmer, J. Bitzer, and C. Marro, "Post-filtering techniques," in *Microphone arrays*, pp. 39–60, Springer, 2001.

[19] Y. A. Huang, A. Luebs, J. Skoglund, and W. B. Kleijn, "Globally optimized least-squares post-filtering for microphone array speech enhancement," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 380–384, IEEE, 2016.

[20] W. Herbordt, *Sound capture for human/machine interfaces: Practical aspects of microphone array signal processing*, vol. 315. Springer Science & Business Media, 2005.

[21] S. Haykin and Z. Chen, "The cocktail party problem," *Neural computation*, vol. 17, no. 9, pp. 1875–1902, 2005.

[22] T.-W. Lee, "Independent component analysis," in *Independent Component Analysis*, pp. 27–66, Springer, 1998.

[23] H. Saruwatari, S. Kurita, K. Takeda, F. Itakura, T. Nishikawa, and K. Shikano, "Blind source separation combining independent component analysis and beamforming," *EURASIP Journal on Applied Signal Processing*, vol. 2003, pp. 1135–1146, 2003.

[24] M. A. Little, “Mathematical foundations of nonlinear, non-gaussian, and time-varying digital speech signal processing,” in *International Conference on Nonlinear Speech Processing*, pp. 9–16, Springer, 2011.

[25] P. Comon, “Independent component analysis, a new concept?,” *Signal processing*, vol. 36, no. 3, pp. 287–314, 1994.

[26] Y. Qi, F. Yao, and M. Yu, “A single-channel ica-r method for speech signal denoising combining emd and wavelet.,” *JCP*, vol. 9, no. 9, pp. 2082–2090, 2014.

[27] P. M. F. Brogueira, “Real time audio system for snr improvement with applications to distant speech recognition,” 2014.

[28] Y. Wang, Ö. Yılmaz, and Z. Zhou, “Phase aliasing correction for robust blind source separation using duet,” *Applied and Computational Harmonic Analysis*, vol. 35, no. 2, pp. 341–349, 2013.

[29] S. C. Douglas, “Introduction to adaptive filters,” *Digital signal processing handbook*, pp. 7–12, 1999.

[30] S. S. Haykin, *Adaptive filter theory.* Pearson Education India, 2008.

[31] M. H. Costa and J. C. M. Bermudez, “A robust variable step size algorithm for lms adaptive filters,” in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 3, pp. III–III, IEEE, 2006.

[32] W.-P. Ang and B. Farhang-Boroujeny, “A new class of gradient adaptive step-size lms algorithms,” *IEEE transactions on signal processing*, vol. 49, no. 4, pp. 805–810, 2001.

[33] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step size,” *IEEE Transactions on Signal Processing*, vol. 41, no. 6, pp. 2075–2087, 1993.

[34] R. Harris, D. Chabries, and F. Bishop, “A variable step (vs) adaptive filter algorithm,” *IEEE transactions on acoustics, speech, and signal processing*, vol. 34, no. 2, pp. 309–316, 1986.

[35] Y. Huang, J. Benesty, and J. Chen, “Optimal step size of the adaptive multichannel lms algorithm for blind simo identification,” *IEEE Signal Processing Letters*, vol. 12, no. 3, pp. 173–176, 2005.

[36] N. D. Gaubitch, M. K. Hasan, and P. A. Naylor, “Generalized optimal step-size for blind multichannel lms system identification,” *IEEE signal processing letters*, vol. 13, no. 10, pp. 624–627, 2006.

[37] V. Malenovsky, Z. Smekal, and I. Koula, "Optimal step-size lms algorithm using exponentially averaged gradient vector," in *Computer as a Tool, 2005. EUROCON 2005. The International Conference on*, vol. 2, pp. 1554–1557, IEEE, 2005.

[38] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and adaptive signal processing: spectral estimation, signal modeling, adaptive filtering, and array processing.* McGraw-Hill Boston, 2000.

[39] C. Schüldt, F. Lindstrom, H. Li, and I. Claesson, "Adaptive filter length selection for acoustic echo cancellation," *Signal Processing*, vol. 89, no. 6, pp. 1185–1194, 2009.

[40] A. Kar and M. Swamy, "Tap-length optimization of adaptive filters used in stereophonic acoustic echo cancellation," *Signal Processing*, vol. 131, pp. 422–433, 2017.

[41] A. H. Sayed, *Fundamentals of adaptive filtering.* John Wiley & Sons, 2003.

[42] P. S. Diniz, *Adaptive filtering.* Springer, 1997.

[43] D. J. Krusienski and W. K. Jenkins, "Particle swarm optimization for adaptive iir filter structures," in *Evolutionary Computation, 2004. CEC2004. Congress on*, vol. 1, pp. 965–970, IEEE, 2004.

[44] P. Regalia, *Adaptive IIR filtering in signal processing and control.* Routledge, 2018.

[45] X. Chen and T. W. Parks, "Design of iir filters in the complex domain," *IEEE transactions on acoustics, speech, and signal processing*, vol. 38, no. 6, pp. 910–920, 1990.

[46] L. Litwin, "Fir and iir digital filters," *IEEE potentials*, vol. 19, no. 4, pp. 28–31, 2000.

[47] B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, J. E. Dong, and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications," *Proceedings of the IEEE*, vol. 63, no. 12, pp. 1692–1716, 1975.

[48] M. H. Hayes, *Statistical digital signal processing and modeling.* John Wiley & Sons, 2009.

[49] F. Beaufays, "Transform-domain adaptive filters: An analytical approach," *IEEE Transactions on Signal processing*, vol. 43, no. 2, pp. 422–431, 1995.

[50] D. P. Ellis, "An introduction to signal processing for speech," *The Handbook of Phonetic Science*, 2008.

[51] A. W. Hull, *Orthogonalization techniques for adaptive filters.* PhD thesis, University of Illinois at Urbana-Champaign, 1994.

[52] M. Dentino, J. McCool, and B. Widrow, "Adaptive filtering in the frequency domain," *Proceedings of the IEEE*, vol. 66, no. 12, pp. 1658–1659, 1978.

[53] S. Narayan, A. Peterson, and M. Narasimha, "Transform domain lms algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 3, pp. 609–615, 1983.

[54] W. K. Jenkins and D. F. Marshall, "Transform domain adaptive filtering," *Digital Signal Processing Handbook*, 2010.

[55] D. Gries, "Texts in computer science," 2005.

[56] D. F. Marshall, W. K. Jenkins, and J. Murphy, "The use of orthogonal transforms for improving performance of adaptive filters," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 4, pp. 474–484, 1989.

[57] Grid Corpus, "The GRID audiovisual sentence corpus," 2013.

[58] T. H. Falk, V. Parsa, J. F. Santos, K. Arehart, O. Hazrati, R. Huber, J. M. Kates, and S. Scollie, "Objective quality and intelligibility prediction for users of assistive listening devices: Advantages and limitations of existing tools," *IEEE signal processing magazine*, vol. 32, no. 2, pp. 114–124, 2015.

[59] A. Almohammedi and M. Deriche, "Variable step-size transform domain ilms and dlms algorithms with system identification over adaptive networks," in *Applied Electrical Engineering and Computing Technologies (AEECT), 2015 IEEE Jordan Conference on*, pp. 1–6, IEEE, 2015.

[60] K. Mayyas, "A transform domain lms algorithm with an adaptive step size equation," in *Signal Processing and Information Technology, 2004. Proceedings of the Fourth IEEE International Symposium on*, pp. 229–232, IEEE, 2004.