Adaptive algorithms and structures with potential application in reverberation time estimation in occupied rooms

Thesis submitted to Cardiff University in candidature for the degree of Doctor of Philosophy.

Yonggang Zhang

Centre of Digital Signal Processing
Cardiff University
2007

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

ABSTRACT

Realistic and accurate room reverberation time (RT) extraction is very important in room acoustics. Occupied room RT extraction is even more attractive but it is technically challenging, since the presence of the audience changes the room acoustics. Recently, some methods have been proposed to solve the occupied room RT extraction problem by utilizing passively received speech signals, such as the maximum likelihood estimation (MLE) technique and the artificial neural network (ANN) scheme. Although reasonable RT estimates can be extracted by these methods, noise may affect their accuracy, especially for occupied rooms, where noise is inevitable due to the presence of the audience. To improve the accuracy of the RT estimates from high noise occupied rooms, adaptive techniques are utilized in this thesis as a preprocessing stage for RT estimation. As a demonstration, this preprocessing together with the MLE method will be applied to extract the RT of a room in which there is significant noise from passively received speech signals. This preprocessing can also be potentially used to aid in the extraction of other acoustic parameters, such as the early decay time (EDT) and speech transmission index (STI).

The motivation of the proposed approach is to utilize adaptive techniques, namely blind source separation (BSS) and adaptive noise cancellation (ANC), based upon the least mean square (LMS) algorithm,

to reduce the noise level contained in the received speech signal, so that the RT extracted from the signal output generated by the preprocessing can be more accurate.

Further research is also performed on some fundamental topics related to adaptive techniques. The first topic is variable step size LMS (VSSLMS) algorithms, which are designed to enhance the convergence rate of the LMS algorithm. The concept of gradient based VSSLMS algorithms is described, and new gradient based VSSLMS algorithms are proposed for applications where the input signal is statistically stationary and the signal-to-noise ratio (SNR) is zero decibels or less.

The second topic is variable tap-length LMS (VTLMS) algorithms. VTLMS algorithms are designed for applications where the tap-length of the adaptive filter coefficient vector is unknown. The target of these algorithms is to establish a good steady-state tap-length for the LMS algorithm. A steady-state performance analysis for a VTLMS algorithm, the fractional tap-length (FT) algorithm is therefore provided. To improve the performance of the FT algorithm in high noise conditions, a convex combination approach for the FT algorithm is proposed. Furthermore, a new practical VTLMS algorithm is also designed for applications in which the optimal filter has an exponential decay impulse response, commonplace in enclosed acoustic environments. These original research outputs provide deep understanding of the VTLMS algorithms.

Finally, the idea of variable tap-length is introduced for the first time into the BSS algorithm. Similar to the FT algorithm, the tap-length of the natural gradient (NG) algorithm, which is one of the most important sequential BSS algorithms is also made variable rather than

fixed. A new variable tap-length NG algorithm is proposed to search for a steady-state adaptive filter vector tap-length, and thereby provide a good compromise between steady-state performance and computational complexity.

The research recorded in this thesis gives a first step in introducing adaptive techniques into acoustic parameter extraction. Limited by the performance of such adaptive techniques, only simulated studies and comparisons are performed to evaluate the proposed new approach. With further development of the associated adaptive techniques, practical applications of the proposed approach may be obtained in the future.

To my loving wife Ning Li

and to my parents

ACKNOWLEDGEMENTS

I would like to give my great thanks to my supervisor Prof. Jonathon A. Chambers, who spent much time consulting with me. Thanks for his great enthusiasm and patience with me. Without his invaluable support and encouragement, this thesis would have not been accomplished. He contributed greatly to every paper I have written, and to every progress step I have made. As I will be a teacher in the future, I have benefited from him so much, not only from his wide knowledge, but also his great enthusiasm with students, and his wonderful teaching skill. He is an example for my future teaching and research career. He has my deepest respect professionally and personally.

I would also like to thank my wife Ning, who accompanied me during the last two years. She is also my best friend and an excellent collaborator. She has contributed to some of my work, and provided a careful proof reading of this thesis. Thanks for her great support to my PhD studies.

My old supervisor in China, Prof. Yanlin Hao, who has supervised me during my master's study, and recommended me to my supervisor, is greatly appreciated. She also provided part of the financial support, which enabled my study in UK. Dr. Saeid Sanei, Dr. Wenwu Wang and Dr. Zhuo Zhang have given much help and many suggestions on not only my research and study, but also my life in Cardiff. As our collab-

orators, Prof. Trevor J. Cox, Dr. Francis F. Li and Mr. Paul Kendrick from Salford University have put many efforts into this project. It has been my great honour to work with them. Prof. Ali H. Sayed has also given very good comments on my work, with whom I have co-authored a paper.

My experience in Cardiff has been wonderful, where I met so many good friends. I would like to thank them: Andrew Aubrey, Clive Cheong Took, Li Zhang, Cheng Shen, Qiang Yu, Yue Zheng, Min Jing, Lay Teen Ong, Kianoush Nazarpour, Jen-Lung Lo and many others. Cardiff has been wonderful for me mainly because of all these friends.

I am grateful to Dr. Rab Nawaz and Dr. Andreas Jakobsson for their guidance with the issues in LaTeX. The major part of the financial support for this research was provided by grants from the Engineering and Physical Sciences Research Council (EPSRC) U.K. The research office staff within the School of Engineering have also been extremely helpful. Finally, I would like to thank my family for their patience and encouragement while we have been separated for many years.

PUBLICATIONS

- Y. Zhang and J. A. Chambers,"A variable tap-length natural gradient blind deconvolution/equalization algorithm," *Electronics Lett.*, accepted, Jun. 2007.

- Y. Zhang, N. Li, J. A. Chambers and A. H. Sayed,"Steady-state performance analysis of a variable tap-length LMS algorithm," *IEEE Trans. Signal Processing*, accepted, Jun. 2007.

- Y. Zhang, J. A. Chambers, P. Kendrick, T. J. Cox, and F. F. Li, "A combined blind source separation and adaptive noise cancellation scheme with potential application in blind acoustic parameter extraction," *Neurocomputing*, submitted, May 2007.

- Y. Zhang, J. A. Chambers, S. Sanei, P. Kendrick, and T. J. Cox, "A new variable tap-length LMS algorithm to model an exponential decay impulse response," *IEEE Signal Processing Lett.*, Vol. 14(4), pp. 263-266, 2007.

- Y. Zhang, and J. A. Chambers, "Convex combination of adaptive filters for variable tap-length LMS algorithm," *IEEE Signal Processing Lett.*, Vol. 13(10), pp. 628-631, 2006.

- Y. Zhang, J. A. Chambers, W. Wang, P. Kendrick and T. J.

Cox, "A new variable step-size LMS algorithm with robustness to nonstationary noise," *ICASSP 2007*, Hawaii, USA, 2007.

- Y. Zhang, J. A. Chambers, F. F. Li, P. Kendrick and T. J. Cox, "A new variable step-size LMS algorithm with robustness to non-stationary noise," *7th International Conference on Mathematics in Signal Processing*, Cirencester, UK, 2006.

- Y. Zhang, N. Li, and J. A. Chambers, "A new gradient based variable step-size LMS algorithm," *8th International Conference on Signal Processing*, Guilin, China, 2006.

- Y. Zhang, J. A. Chambers, P. Kendrick, T. J. Cox and F. F. Li, "Blind estimation of reverberation time in occupied rooms," *10th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, KES 2006*, Bournemouth, UK, 2006.

- Y. Zhang, J. A. Chambers, F. F. Li, P. Kendrick and T. J. Cox, "Acoustic parameter extraction from occupied rooms utilizing blind source separation," *EUSIPCO2006*, Florence, Italy, 2006.

- P. Kendrick, T.J. Cox, J. Chambers, Y. Zhang and F. F. Li, "Reverberation time parameter extraction form music signals using machine learning techniques," *5th International Postgraduate Research Conference in The Built and Human Environment*, Salford, UK, 2006.

- P. Kendrick, T. J. Cox, Y. Zhang, J. A. Chambers and F. F. Li, "Reverberation time parameter extraction from music signals using machine learning techniques," *ICASSP 2006*, Toulouse, France, 2006.

- T. J. Cox, P. Kendrick, Y. Zhang, J. A. Chambers and F. F. Li, "Extracting room acoustic parameters from music," *6th International Conference on Auditorium Acoustics*, Copenhagen, Denmark, 2006.

- P. Kendrick, F. F. Li, T. J. Cox, Y. Zhang, and J. A. Chambers, "Blind estimation of room reverberation parameters: improving the accuracy of the maximum likelihood approach," Acta Acustica, accepted with minor changes, 2007.

Acronyms

AMUSE	Algorithm for Multiple Unknown Signals Extraction
ANC	Adaptive Noise Cancellation
ANN	Artificial Neural Network
BPF	Band Pass Filter
BSS	Blind Source Separation
DFT	Discrete Fourier Transform
DOA	Direction Of Arrival
EDT	Early Decay Time
EMSE	Excess Mean Square Error
FIR	Finite Impulse Response
FT	Fractional Tap-length
IDFT	Inverse Discrete Fourier Transform
IIR	Infinite Impulse Response
LMS	Least Mean Square
MLE	Maximum Likelihood Estimation

MSD	Mean Square Deviation
MSE	Mean Square Error
NG	Natural Gradient
NLMS	Normalized LMS
PDF	Probability Density Function
RT	Reverberation Time
SNR	Signal-to-Noise Ratio
SOBI	Second-Order Blind Identification
STI	Speech Transmission Index
VSSLMS	Variable Step Size LMS
VTLMS	Variable Tap-length LMS

CONTENTS

List of Figures

List of Tables

Chapter 1

INTRODUCTION

Room reverberation time (RT) is a very important acoustic parameter for characterizing the quality of an auditory space. The estimation of room RT has been of interest to engineers and acousticians for nearly a century. This parameter is defined as the time taken by a sound to decay 60dB below its initial level after it has been switched off [2]. Reverberation time by this classical definition is referred to as RT60. For convenience of presentation, it will be denoted as T_{60} throughout the thesis.

The reverberation phenomenon is due to multiple reflections of the sound from surfaces within a room. It distorts both the envelope and fine structure of the received sound. Room RT provides a measure of the listening quality of a room; so obtaining an accurate room RT is very important in acoustics. From an application perspective, obtaining accurate room acoustic measures such as room RT is often the first step in applying existing knowledge to engineering practices, diagnosing problems of spaces with poor acoustics and proposing remedial measures. From a scientific research perspective, more realistic and accurate measurements are demanded to enrich the existing knowledge base and correct its imperfections [3].

Many methods have been proposed to estimate RT. Typical meth-

ods can be seen in [4], [5], [6], [7] and [8]. The first RT estimation method proposed by W. C. Sabine is introduced in [4]. It utilizes the geometrical information and absorption characteristics of the room. The methods presented in [5] and [6] extract RT by recording good controlled excitation signals, and measuring the decay rate of the received signal envelope. These traditional methods are not suitable for occupied rooms, where prior information of the room or good controlled excitation signals are normally difficult to obtain. In order to measure the RT of occupied rooms, an artificial neural network (ANN) method is proposed in [7], and a maximum likelihood estimation (MLE) based scheme is proposed in [8]. Both of them utilize modern digital signal processing techniques, and can extract RT from occupied rooms by only utilizing passively received speech signals (throughout this thesis RT estimation is assumed to be "in-situ", i.e., the excitation signal is assumed to be one generated by someone already within the room, methods based on introducing an external source are not considered). The advantage of these methods is that no prior information of the room or good controlled excitation signals are necessary. However, their performance will be degraded and generally biased by noise, thus they are not suitable for high noise conditions. In this thesis the term high noise is used to denote signal-to-noise ratios (SNR) of approximately 0dB and below.

To solve the high noise occupied room RT estimation problem, a new approach is proposed in this thesis, which utilizes a combination of blind source separation (BSS) and adaptive noise cancellation (ANC) based upon the least mean square (LMS) algorithm. These adaptive techniques are exploited in a preprocessing stage before the RT estima-

tion. The aim of this study is to improve the accuracy of the existing occupied room RT estimation methods by reducing the unknown noise contained in the received speech signals. Fundamentally new research on the adaptive techniques is also provided.

In this chapter, traditional room RT estimation methods proposed in [4], [5] and [6] are firstly introduced, followed by the description of occupied room RT estimation methods in [7] and [8]. The background and motivation of the new occupied room RT estimation technique is then provided. On the basis of this proposed approach, the scope of the study within this thesis is given, which concentrates mainly on the topics of variable step size LMS (VSSLMS) algorithms, variable tap-length LMS (VTLMS) algorithms, and a new variable tap-length natural gradient (NG) algorithm. Finally, the organization of the thesis is provided.

1.1 Traditional RT estimation methods

1.1.1 Geometrical based method

An early RT estimation method is proposed by W. C. Sabine [2] and presented in [4]. This method is derived under the condition that the sound energy in a room is uniformly distributed.

According to the formulation in [4], if a sound source $s(t)$ radiates into a room, where t denotes continuous time, its power can be balanced by the variation of the energy content Vw, where V is the volume of the room in m^3 and w is the sound energy density in J/m^3, and also by the losses due to the absorption of the room boundary which has

the absorption coefficient α

$$E\{s^2(t)\} = V\frac{dw}{dt} + B\alpha S \tag{1.1.1}$$

where $\frac{dw}{dt}$ denotes the differential of w with respect to t, S is the total surface area of the room in m^2, and B is the irradiation strength to the surface walls in units W/m^2

$$B = \frac{c}{4}w \tag{1.1.2}$$

where c is the velocity of the sound in air, $331m/s$. After the sound is switched off, i.e., $s(t) = 0$ for $t > 0$, substituting (1.1.2) into (1.1.1) the solution for the energy density is obtained

$$w(t) = w_0 e^{-2t/\tau} \qquad t > 0 \tag{1.1.3}$$

where w_0 is the sound energy density when the source sound is switched off, and τ is the decay constant in seconds

$$\tau = \frac{8V}{cS\alpha} \tag{1.1.4}$$

For convenience of presentation, the discrete formulation of signals and parameters will be used for the remainder of this thesis. Assuming the sampling frequency of signals is Fs, i.e., the sampling period is $T = \frac{1}{Fs}$, and the unit of time will be the sample period rather than seconds. The discrete formulation of the energy density is

$$w(n) = w_0 e^{-2n/\tau} \qquad n > 0 \tag{1.1.5}$$

where n denotes the sample index and τ is the decay constant in sample periods

$$\tau = \frac{8VFs}{cS\alpha} \tag{1.1.6}$$

Note that the theoretical value $E\{y_T^2(n)\}$ of the received sound sequence $y_T(n)$ is proportional to the energy density $w(n)$, $E\{y_T^2(n)\} = w(n)K$ where K is an unknown constant. Substituting (1.1.5) into this formulation yields

$$E\{y_T^2(n)\} = E\{y_T^2(0)\}e^{-2n/\tau} \tag{1.1.7}$$

where $E\{y_T^2(0)\}$ is corresponding to w_0. The natural logarithm of $E\{y_T^2(n)\}$ can then be formulated as

$$\ln[E\{y_T^2(n)\}] = \ln[E\{y_T^2(0)\}] + \frac{-2n}{\tau} \tag{1.1.8}$$

It is clear to see from (1.1.8) that the logarithm of $E\{y_T^2(n)\}$ has the form of a straight line with respect to the sample index. From the definition of RT the following equation is obtained

$$10\log_{10}\frac{E\{y_T^2(T_{60} * Fs)\}}{E\{y_T^2(0)\}} = -60 \tag{1.1.9}$$

Substituting (1.1.8) into (1.1.9) the RT can be calculated as

$$T_{60} = 6.91\tau \tag{1.1.10}$$

This is a very important formulation of the relationship between the decay constant and the RT which is used in most RT estimation methods. For the geometrical based method, the decay constant is calculated

from (1.1.6), and the RT can then be obtained directly from (1.1.10). This method has been widely used in anechoic chamber measurements, design of concert halls, classrooms, and other acoustic spaces where the quality of the received sound is of importance [4].

1.1.2 Interrupted noise method

This method is formulated in [5]. According to the definition, room RT can be determined from decay curves obtained by radiating sound into test rooms. The sound source is switched on, and switched off when the received sound energy reaches a steady-state. When the sound source is switched off, the decay curve of the received sound energy is recorded. Assuming the excitation noise signal $s(n)$ is statistically stationary and white, and $h(n)$ is the room impulse response, a single realization of the received sound signal $y(n)$ from the interrupted noise method can be described as

$$y(n) = \sum_{\tau=-\infty}^{0} s(\tau)h(n-\tau) \tag{1.1.11}$$

where the excitation signal is assumed to be switched off at $n = 0$. The lower limit of the sum, $-\infty$, indicates that the excitation noise signal has been switched on for a sufficiently long time, so that when it is switched off, the sound level has reached its steady-state. Since the decay curve obtained from the excitation noise signal will be different from trial-to-trial due to the random fluctuations of the signal, many experiments are needed to obtain a set of decay curves, and the averaged received signal decay curve, denoted as $\overline{y(n)}$, is used to extract the RT.

Similar to that in the geometrical based method in (1.1.7), the exponential decay model is also used in the value $\overline{y^2(n)}$. With a *polyfit* operation, the decay rate parameter τ can be estimated from the nat-

ural logarithm of $\overline{y^2(n)}$. (The function *polyfit* can be found in Matlab, and is used to find the best fit damping decay constant τ from the logarithm of $\overline{y^2(n)}$ in a least-squares-sense). Then the RT can be determined by equation (1.1.10).

1.1.3 Schroeder's method

Schroeder's method is also called the integrated impulse response method, or backward integration method [6]. By using equation (1.1.11), Schroeder establishes a relationship between the mean squared average of the decay curve $y(n)$ and the impulse response $h(n)$. In this method, a smoothed decay curve is produced by backward integration summation of the impulse response $h(n)$

$$E\{y^2(n)\} = \sigma^2 \sum_{\tau=n}^{\infty} h^2(\tau) \tag{1.1.12}$$

where σ^2 is the power of the excitation signal, assumed to be of impulsive nature, such as a pistol shot or a hand-clap, which is used to excite the impulse response of the room. A smoothed decay curve $E\{y^2(n)\}$ is obtained from equation (1.1.12), and the decay rate constant τ can be extracted from $E\{y^2(n)\}$ by using the *polyfit* operation, similar to that in the interrupted noise method. The room RT can thereby be obtained. As compared with the interrupted noise method, in which many experiments are needed to obtain an averaged decay curve, the advantage of Schroeder's method is that only a single measurement of the room impulse response is needed.

Although all the above methods have been used successfully in many applications, they are not suitable for occupied rooms, mainly because

of the following reasons:

(1) In many applications, both the room geometry and the absorptive characteristics are difficult to obtain, especially for occupied rooms, where the presence of the audience changes the room acoustic characteristics. This limits the application of the geometrical based method for occupied rooms.

(2) In both the interrupted noise method and Schroeder's method, high sound pressure noises are used as excitation signals. However, for the audience, exposure to loud noise for a long period is difficult or impractical.

(3) Maintaining the required SNR is another technical obstacle in occupied measurements. The presence of the audience inevitably aggravates the noise and consequently reduces measurement accuracy.

In summary, to estimate the room RT for an occupied room, the presence of the audience must be considered.

1.2 Occupied room RT estimation methods

Some modern digital signal processing techniques have been utilized to estimate occupied room RT. One of the occupied room RT estimation methods is proposed in [3] and [7], where the RT is extracted by using an ANN approach, and another is proposed in [8], in which an MLE method is utilized.

1.2.1 ANN method

According to the formulation in [3], the application of the ANN method for RT estimation can be divided into two stages: the training stage, as shown in Fig.1.1, and the operational stage as shown in Fig.1.2.

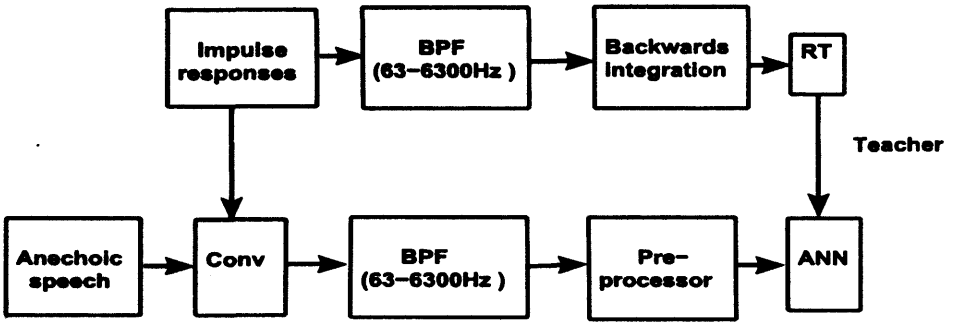

Figure 1.1. The training stage of the ANN method

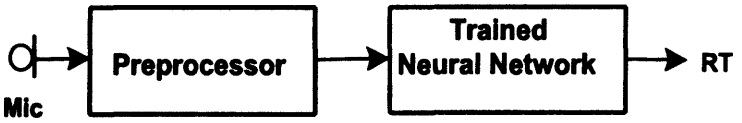

Figure 1.2. The operational stage of the ANN method

In the training stage, a large number of anechoic speech examples are convolved with known room impulse responses to generate training samples. These training samples are firstly passed through a band pass filter (BPF) with a range of $63 - 6300$Hz according to the speech spectrum bandwidth [3]. This range of frequencies only pertains to this ANN method and is not adopted in the processing proposed in this thesis. Then a preprocessor is used to normalize the band filtered signals, and transfer them to suitable ANN inputs, such as the short-time average root-mean-square values. The impulse responses are band pass filtered first, then the backward integration method, i.e., Schroeder's

method is performed to calculate the RTs from these impulse responses. With these calculated values of RTs, the ANN is trained to learn the nonlinear relationship between the input and real RTs, as such the ANN is performing functional approximation.

Following the training stage, an operational stage can then be performed. Similar to as in the training stage, the recorded speech signals are passed through the preprocessor, and the output of the preprocessor is used as input to the trained ANN. Since the nonlinear relationship between RTs and the inputs has been setup in the training stage, with a new input, the corresponding RT can then be found by the trained ANN.

As shown by the authors, reasonable RTs can be extracted by the ANN method, but the generalization performance of the network is very much dependent upon the richness of the training data.

1.2.2 MLE method

The MLE method is also based upon utilizing a passively received speech signal to extract room RT. In this method, the received speech signal is divided into a set of overlapped segments. An observed signal vector can be obtained from each segment. With an exponentially damped Gaussian white noise model, the decay rate of each segment envelope can be obtained by using an MLE approach. One RT estimate can be calculated from the decay rate, and a series of RT estimates can be obtained from the whole passively received speech signal. All these estimates are accumulated in a histogram, and the most likely RT is identified as the first dominant peak of the histogram. As compared with the ANN method, the MLE method is easier to implement since

no training stage is needed.

Although both methods have been demonstrated to be good RT estimation methods for certain occupied rooms by only utilizing passively received speech signals, their performance will be degraded and generally biased if the noise level is high. Therefore, both methods are limited by the noise level.

To make the occupied room RT estimation methods more robust and accurate, an intuitive way is to remove the unknown noise signal from the received speech signal as much as possible before the RT estimation. This is also the motivation of the work in this thesis. Since as compared with the ANN approach, the MLE method is easier to implement and not limited by extensive available training data, the study for the remainder of this thesis will be based on the MLE method. A detailed introduction of the MLE method can be found in the next chapter.

1.3 Background and motivation of the proposed approach

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of the U.K. funded project "Room Acoustics Parameters from Music" under grant number GR/S77530/01, which was proposed by Prof Jonathon A. Chambers from the Centre of Digital Signal Processing, Cardiff University, together with Prof Trevor J. Cox, Dr Francis F. Li, and Mr Paul Kendrick from the School of Acoustic and Electronic Engineering, University of Salford. This project followed a previous EPSRC project namely "Quantifying Room Acoustic Quality Using Artificial Neural Networks" under grant number GR/L34396 and completed by Prof Trevor J. Cox and Dr Francis F. Li. In that

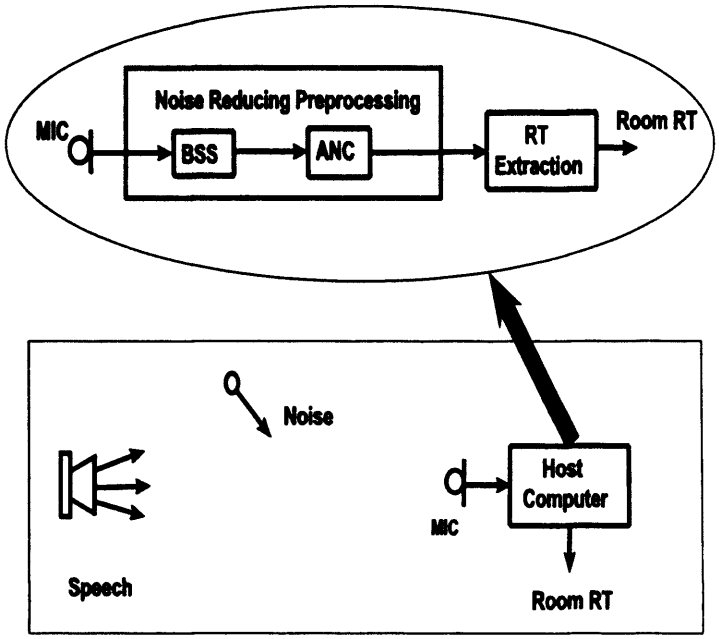

Figure 1.3. RT estimation from high noise occupied rooms

project, a neural network approach was setup to extract room acoustic parameters from passively received speech signals, as introduced in the previous section, and can be seen in [3] and [7].

As a part of the project "Room Acoustics Parameters from Music", the object of the work in this thesis is to utilize adaptive techniques to improve the accuracy of RT estimates in high noise conditions. If RT can be reliably estimated from high noise speech signals, a similar approach can be potentially performed on passively received music signals, and for estimating other acoustic parameters such as the early decay time (EDT) and speech transmission index (STI).

Since for the passively received speech signal, both the prior knowledge of the noise component and the excitation speech component are unknown, some blind signal processing approaches are necessary. A

powerful tool for extracting some unknown noise interference signal from a mixture of speech signals is the convolutive BSS method [9] and [10]. Naturally, given two spatially distinct observations, BSS can attempt to separate the mixed signals to yield two independent signals. One of these two signals consists mainly of the excitation speech signal plus residue of the noise and the other signal contains mostly the noise. The estimated noise signal obtained from BSS then serves as a reference signal within an ANC, in which an LMS algorithm is utilized. The output of the ANC is a reverberant speech signal with reduced noise component. In this work it is assumed that the ANC only locks onto the contaminated noise component. As will be shown by the simulations, the accuracy of RT obtained from the output of the ANC is improved, due to the noise reducing preprocessing. Different stages of this framework are shown in Fig.1.3.

Although the performance of the proposed approach depends on the performance of BSS, and the noise is modelled as directional noise, the idea of noise reducing preprocessing can be potentially extended to more complex models with the concomitant development of more powerful BSS algorithms.

1.4 Scope of this study

In this study, a new approach for high noise occupied room RT estimation is proposed. As byproducts of this study, further research on adaptive techniques is also performed. The main contributions of this thesis can be summarized as follows:

1. A noise reducing approach by utilizing BSS and ANC is introduced into the RT estimation problem to improve the accuracy of RT

estimates.

2. The concept of gradient based VSSLMS algorithms is proposed. New VSSLMS algorithms designed for applications with high level (typically SNR=0dB or below) statistically stationary noise or nonstationary noise are proposed to accelerate the convergence of the LMS algorithm which is used in the ANC stage. In this thesis only stochastic gradient type algorithms are considered due to their well-known lower computational complexity, better convergence and tracking performance in low, typically 0dB SNR environments [11].

3. New research results have been obtained for VTLMS algorithms, which are designed to search for a good choice of the steady-state adaptive filter tap-length. Although many VTLMS algorithms have been proposed in recent years, the fractional tap-length (FT) algorithm proposed in [12] has been shown to be more robust as compared with other methods. The contributions of the work concerned with VTLMS algorithms are as follows: a steady-state performance analysis of the FT algorithm; improvement of the convergence performance of the FT algorithm in a high noise condition by utilizing a convex combination approach; and a new practical variable tap-length LMS algorithm for applications in which the optimal filter has an exponential decay impulse response. These research results are potentially very useful in many applications where the optimal tap-length of the LMS algorithm is unknown.

4. The idea of variable tap-length is introduced for the first time into the BSS research area, in particular for a key sequential BSS algorithm, the NG algorithm. A variable tap-length NG algorithm is proposed to search for a good choice of the adaptive tap-length, which

provides a good trade-off between the steady-state performance and computational complexity. Due to the similarity of the optimal tap-length model between the LMS algorithm and the BSS algorithms, more research is required for variable tap-length BSS algorithms.

1.5 Organization of the thesis

This thesis is divided into seven chapters:

Following the introduction chapter, the MLE based RT estimation method is described in detail in Chapter 2. A background introduction for the adaptive schemes, including the LMS algorithm and BSS techniques, is provided in Chapter 3. To solve the high noise occupied room RT estimation problem, a new framework using BSS and ANC is proposed in Chapter 4. Further research on BSS and ANC is given in the remaining chapters. In Chapter 5, VSSLMS algorithms are discussed and the concept of gradient based VSSLMS algorithms is described. New gradient based VSSLMS algorithms with robustness to statistically stationary or nonstationary noise signals are also proposed. Research on VTLMS algorithms is performed in Chapter 6, where the FT algorithm is introduced first, and a steady-state performance analysis of the FT algorithm follows. To improve the performance of the FT algorithm in high noise environments, a new convex combination approach for the FT algorithm is provided. A practical variable tap-length algorithm is also proposed for applications in which the optimal filter has an exponential decay impulse response. In Chapter 7, the NG algorithm for blind deconvolution is introduced, and a new variable tap-length NG algorithm is presented. Chapter 8 provides the summary of this thesis and gives suggestions for future work.

Chapter 2

MAXIMUM LIKELIHOOD ESTIMATION BASED ROOM RT ESTIMATION METHOD

In this chapter a detailed introduction to the maximum likelihood estimation (MLE) based room reverberation time (RT) estimation method [8] is given.

In this method, the RT of an occupied room is extracted from a passively received speech signal. At first, the passively received speech signal is divided into several overlapped segments with the same length. Each segment can be deemed as an observed vector. This observed vector is modelled as an exponentially damped Gaussian random sequence, i.e., it is modelled as an element-by-element product of two vectors, one is a vector with an exponentially damped structure, and the other is composed of independent identical distributed (i.i.d.) Gaussian random samples. Note that the exponentially damped vector also models the envelope of the speech segment. The MLE approach is applied to the observed vector to extract the decay rate of its envelope. The RT can then be easily obtained from the decay rate, according to its definition. An estimation of RT can be extracted from one segment, and a series of

RT estimates can be obtained from the whole passively received speech signal. The most likely RT of the room can then be identified from these estimates.

In this chapter, the exponentially damped Gaussian random sequence model is introduced first, and then the MLE method extracting a RT estimate from one segment of speech signal is described. Two methods for calculating the decay rate of the speech segment envelope in the MLE approach are provided. One is an online method, the other is a block-based method. Based on the MLE approach, identifying the most likely RT from a passively received speech signal is finally presented. The MLE based RT estimation method introduced in this chapter will be used together with the proposed noise reducing preprocessing in Chapter 4 to show the advantage of the proposed preprocessing in a high noise environment.

2.1 Exponentially damped Gaussian white noise model

In the MLE based RT estimation method, an exponentially damped Gaussian white noise model is utilized. This model is motivated by recorded room responses generated from some impulsive signals, such as a hand-clap or a pistol shot. In the recorded response, there is a direct sound due to the delay induced by the transmission path, followed by a series of early reflections, and then a reverberant tail appears, which consists of dense reflections due to multiple scattering. Unlike the direct sound and early reflections, acoustic reverberation consists of a fine structure that can be described only statistically. Usually, the fine structure is considered to be an uncorrelated random process. However, the decaying envelope is a deterministic signal parameterized

by a time-constant, which is linearly proportional to the reverberation time. Based on the above characteristics of the reverberant tails, a convenient and highly simplified model is to consider the reverberation tail to be an exponentially damped uncorrelated noise sequence with Gaussian characteristics [8]:

$$y(n) = a(n)x(n) \tag{2.1.1}$$

where $y(n)$ is the observed reverberant tail signal, $x(n)$ is an i.i.d. random sequence with a normal distribution $N(0, \sigma)$ where 0 is the zero mean value and σ is the standard deviation, $a(n)$ is a time-varying term which formulates the decay envelope of $y(n)$. From this model it is apparent that the samples from $y(n)$ are independent but not identically distributed. The probability density function (PDF) of $y(n)$ is $N(0, \sigma a(n))$. That is, the sequence $a(n)$ modulates the instantaneous energy of the fine structure of the speech segment. Denoting the damping rate of the sound envelope by a single decay rate τ, the sequence $a(n)$ can be uniquely determined by

$$a(n) = \exp(-n/\tau) \tag{2.1.2}$$

Thus sequence $a(n)$ can be replaced by a scalar parameter a

$$a(n) = a^n \tag{2.1.3}$$

where

$$a = \exp(-1/\tau) \tag{2.1.4}$$

Substituting (2.1.3) into (2.1.1), the observed sequence $y(n)$ can be

modelled as

$$y(n) = a^n x(n) \tag{2.1.5}$$

With a set of observed signal samples, the MLE approach can be applied to extract the estimates of both the parameter a and the parameter σ. The decay parameter τ can then be obtained from (2.1.4). As introduced in the previous chapter, the RT can be directly calculated from the decay rate parameter $T_{60} = 6.91\tau$.

2.2 MLE method to extract the decay rate constant

The MLE approach can be used to extract some unknown parameters from a set of observed samples. This approach has many attractive attributes. First, it has good convergence properties as the number of observed signal samples increases. Furthermore, it is generally simpler than alternative methods, such as Bayesian techniques [13].

Denote the N-dimensional vector of the observed signal samples $y(n)$ by $\mathbf{y}$, the likelihood function of $\mathbf{y}$ from the model described in (2.1.5) can be formulated as [8]:

$$L(\mathbf{y}; a, \sigma) = \left(\frac{1}{2\pi a^{(N-1)}\sigma^2}\right)^{\frac{N}{2}} \times \exp\left(-\frac{\sum_{n=0}^{N-1} a^{-2n} y^2(n)}{2\sigma^2}\right) \tag{2.2.1}$$

where a and σ are unknown parameters to be estimated from the observation $\mathbf{y}$. The log-likelihood function is

$$\ln L(\mathbf{y}; a, \sigma) = -\frac{N(N-1)}{2}\ln(a) - \frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=0}^{N-1} a^{-2n} y^2(n) \tag{2.2.2}$$

Thus for a given observation window N and observed signal vector $\mathbf{y}$, the log-likelihood function is determined by parameters a and

σ. These two parameters can be estimated using an MLE approach. Differentiating the log-likelihood function in (2.2.2) with respect to a and σ yields

$$\frac{\partial \ln L(\mathbf{y}; a, \sigma)}{\partial a} = -\frac{N(N-1)}{2a} + \frac{1}{a\sigma^2}\sum_{n=0}^{N-1} n a^{-2n} y^2(n) \tag{2.2.3}$$

and

$$\frac{\partial \ln L(\mathbf{y}; a, \sigma)}{\partial \sigma} = -\frac{N}{\sigma} + \frac{1}{\sigma^3}\sum_{n=0}^{N-1} a^{-2n} y^2(n) \tag{2.2.4}$$

By setting the partial derivatives of the log-likelihood function (2.2.3) and (2.2.4) to zero, the MLE estimates of a and σ can be obtained from the following equations

$$-\frac{N(N-1)}{2a} + \frac{1}{a\sigma^2}\sum_{n=0}^{N-1} n a^{-2n} y^2(n) = 0 \tag{2.2.5}$$

and

$$\sigma = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} a^{-2n} y^2(n)} \tag{2.2.6}$$

Note that both second derivatives of the log-likelihood functions with respect to a and σ are less than zero [8], thus the solutions of (2.2.5) and (2.2.6) maximize the log-likelihood function. The problem of RT estimation from an observed vector now transfers to two equations, which will be solved in the following section.

2.3 Calculation methods for the MLE approach

In this section, two methods are introduced to calculate the parameter a from equations (2.2.5) and (2.2.6). One of them is an online method, and the other is a block-based method [14].

2.3.1 The online calculation method

According to equation (2.1.4) the parameter a is known to fall in the range $a \in [0, 1)$. In the online method, the range of parameter a is quantized into Q values, represented by $a_j, j = 1, ..., Q$. For each a_j, substituting (2.2.6) into (2.2.2), the log-likelihood function with respect to a_j becomes [14]

$$\ln L(\mathbf{y}; a_j) = -\frac{N}{2}\{(N-1)\ln(a_j) - \ln[\frac{2\pi}{N}\sum_{n=0}^{N-1} a_j^{-2n} y^2(n)] - 1\} \quad (2.3.1)$$

The solution of a is determined as

$$a = \arg\max_{a_j}\{\ln L(\mathbf{y}; a_j)\} \quad (2.3.2)$$

The motivation of this quantized approach is that for most applications, it is not necessary to determine the decay time-constant to arbitrary precision [14]. The computation of $\ln L(a_j; \mathbf{y})$ can be performed in an online way as follows. By defining

$$g(n) = \beta^{N-1} \sum_{r=n-N+1}^{n} \beta^{r-n} y^2(r) \quad (2.3.3)$$

where $\beta = a_j^{-2}$, the log-likelihood function with the currently given observed vector $\mathbf{y}_n$ with respect to a_j can be formulated as

$$\ln L(\mathbf{y}_n; a_j) = -\frac{N}{2}\{(N-1)\ln(a_j) - \ln g(n) - 1\} \quad (2.3.4)$$

where $\mathbf{y}_n$ is the vector which contains the samples [y(n-N+1),...,y(n)],

and the recursive update rule for $g(n)$ is

$$g(n+1) = \beta^{-1}[g(n) + \beta^N y^2(n+1) - y^2(n+1-N)] \qquad (2.3.5)$$

For each a_j, with a calculated estimation of the log-likelihood function $\ln L(\mathbf{y}_n; a_j)$, the next log-likelihood function $\ln L(\mathbf{y}_{n+1}; a_j)$ can be updated using (2.3.4) together with the update of $g(n)$ in (2.3.5). For Q bins, this online method requires $5Q$ MULS, $5Q$ ADDS, and the evaluation of Q logarithms [14]. Since the parameter a is an exponential formulation of the decay constant τ, it is quantized logarithmically, so that the corresponded decay constant τ is quantized linearly. To choose the number of bins Q, the only information needed is whether the reverberation is high, moderate or low. A recommendation is that one bin can be used for very high ($> 10s$) and one bin for low RT values ($< 0.01s$). Estimates falling within these bins may be rejected, since the estimates $T_{60} > 10$ indicate that the algorithm is tracking a region that may not be a decay, and estimates $T_{60} < 0.01$ suggest open-air or anechoic conditions. For the values between these extremes, 5-6 bins can be selected. Thus, the number of bins that are required may not exceed 10 [14].

The advantage of this online method is that an online histogram in pre-determined bins can be constructed, and the calculation results from the previous observed vector can be used in the next iterative calculation, which speeds up the estimation process.

Note that the online algorithm is run at intervals of one sample, i.e., the next observed vector is obtained by advancing the previous observed vector one sample, due to the update rule in (2.3.4) and (2.3.5).

2.3.2 The block-based calculation method

The block-based algorithm is to perform a single estimate of the parameter a with an observed vector $\mathbf{y}$. This algorithm is a gradient based algorithm, and designed to search for the solution of equation (2.2.5) in an iterative way. Substituting (2.2.6) into (2.2.3) yields the gradient of the log-likelihood function with respect to a:

$$\frac{\partial \ln L(\mathbf{y};a)}{\partial a} = \frac{N}{a}\{-\frac{N-1}{2} + \frac{\sum_{n=0}^{N-1} n a^{-2n} y^2(n)}{\sum_{n=0}^{N-1} a^{-2n} y^2(n)}\} \tag{2.3.6}$$

The update rule for solving equation (2.2.5) with respect to a is

$$a_{k+1} = a_k - \mu \frac{\partial \ln L(\mathbf{y};a)}{\partial a} \tag{2.3.7}$$

where μ is a fixed parameter which controls the speed of the convergence.

For the block-based method, the calculation of the parameter a with a given observed vector is independent of the previous calculation, thus the choice of the interval between neighboring observed vectors is more flexible.

In most applications, the online method is preferred, because as compared with the block-based method, it can not only generate online RT estimations, which is very useful in many applications, but also generally has a lower computational complexity. Furthermore, the choice of the step size μ in (2.3.7) in the block-based method is difficult to analyze. Based on the above consideration, the online method will be utilized in the application of MLE based RT estimation method for the remainder of this thesis.

2.4 RT extraction from a running speech signal

With a passively received speech signal $y(n)$, the MLE approach proceeds by sliding the window of length N over $y(n)$ with intervals determined by the step size of the advancing window. This process produces a series of estimates of a and all these estimates are accumulated in a histogram.

Since the estimates of a track the sound decay, some of them are obtained during a free decay following the cessation of a sound segment, while others are obtained when the sound is ongoing. Two kinds of observed vectors $\mathbf{y}$ will make the model fail: $\mathbf{y}$ is not in the free decay period or $\mathbf{y}$ is in the free decay period, but initiated by a sound with a gradual rather than rapid offset. In the former case, the parameter a will likely be implausible, due to the wide fluctuations of the ongoing speech signal. In the latter case, the resulting estimates of a will always be larger than the true room RT. The reason is that they are obtained from a signal which is generated by a gradual offset decay signal convolved with the room impulse response, which biases the extracted decay rate parameter to be larger than the real decay rate parameter.

By considering the above two failure cases, a simple and intuitive way for selecting a from a set of estimates is to choose the dominant peak value of a from its histogram at the lower end of the range [8].

In practice, the performance of an MLE based method depends on the choice of the window length N. Although the window length in the model should be long enough to cover the decay period, it is limited by the duration and occurrence of gaps between sound segments. As discussed in [8], window lengths around $N = 4\tau * Fs$ are good choices in practice. This criterion will also be used in the simulations in Chapter

4.

As can be seen in [8], several simulations of the MLE method have been performed, in which room RTs are extracted from broadband white noise bursts, isolated speech words, and connected speech signals. In all these simulations, the MLE based RT estimation method provides reasonable results in noise free environments.

2.5 Conclusion

In this chapter, a detailed introduction of the MLE based RT estimation method is provided. The great advantage of this method is that, as compared with other techniques, only the passively received speech signal is needed, which is particularly useful for occupied room RT estimations. The room RT is obtained by utilizing the envelope of the free decay speech segment. Although the free decay segments are unknown, with the advancing of a window, a series of estimates of the RT will accumulate on certain values, and the first dominant peak of the histogram is identified as the most likely room RT. It is straightforward to see that the performance of this approach still depends on the noise level. When the noise level is high, the fine structure of the free decay speech segment will be contaminated, and the estimated RT will be generally biased, as can be seen in the simulations in Chapter 4. This method will be used in Chapter 4, together with the noise reducing preprocess to extract the RTs from high noise environments.

Chapter 3

BACKGROUND INTRODUCTION TO ADAPTIVE TECHNIQUES

As can be seen in the first chapter, the adaptive least-mean-square (LMS) algorithm and the blind source separation (BSS) scheme are utilized in the proposed RT estimation framework. In this chapter a background introduction to both adaptive techniques is given. The material included in this chapter also provides a basis for the remaining chapters.

This chapter is organized as follows: the LMS algorithm is introduced in Section 3.1. The BSS problem is formulated in Section 3.2. Instantaneous BSS algorithms are described in Section 3.3. Convolutive BSS algorithms are presented in Section 3.4. The permutation issue which occurs in frequency domain convolutive BSS algorithms is discussed in Section 3.5. Section 3.6 provides the conclusion.

3.1 Introduction to the LMS algorithm

The LMS algorithm can be derived from the Wiener filter, which is a finite impulse response (FIR) filter that minimizes the mean-square-error (MSE) between the desired signal and its linear estimate obtained from another reference signal. The Wiener filter is a theoretical and ideal solution, and its design needs a priori information about the statistics of the input and desired signals. To avoid the matrix inverse operation in the Wiener filter, the steepest descent algorithm can be utilized which is designed to converge to the Wiener solution in an iterative manner. Furthermore, since the statistical values of the signals are unavailable in some situations, the LMS algorithm is derived, in which the statistical values are replaced with their instantaneous estimates. This algorithm has been extensively used in many applications as a consequence of its simplicity and robustness [15] and [16].

3.1.1 The FIR Wiener filter

The Wiener filter is an optimum filter in the least MSE sense, i.e., it can minimize the MSE value of the error signal which is defined as the difference between the desired signal and its estimate. The model of the Wiener filter is as follows:

It is assumed that the input signal $x(n)$ and the desired signal $d(n)$ are jointly wide-sense stationary and zero-mean signals, n denotes the discrete time index, and their relationship can be formulated as

$$d(n) = \mathbf{w}_{opt}^T \mathbf{x}(n) + t(n) \tag{3.1.1}$$

where $\mathbf{w}_{opt}$ is an unknown optimal filter coefficient vector with a tap-

length of L, $(\cdot)^T$ denotes the transpose operation, $\mathbf{x}(n) = [x(n), ..., x(n-L+1)]^T$, $t(n)$ is the unknown noise signal and uncorrelated with $x(n)$. The error between the desired signal and its estimate is defined as

$$e(n) = d(n) - \mathbf{w}^T\mathbf{x}(n) \tag{3.1.2}$$

where $\mathbf{w}$ is a coefficient vector to model the unknown optimal filter $\mathbf{w}_{opt}$. The MSE is then defined as $E\{e^2(n)\}$, and the least MSE is obtained when $\mathbf{w} = \mathbf{w}_{opt}$. Using (3.1.2) the MSE can also be formulated as

$$E\{e^2(n)\} = E\{[d(n) - \mathbf{w}^T\mathbf{x}(n)]^2\} \tag{3.1.3}$$

The Wiener filter is designed to find the filter coefficient vector $\mathbf{w} = \mathbf{w}_{opt}$ which minimizes the MSE. It is straightforward to see from (3.1.3) that $E\{e^2(n)\}$ is a quadratic function of $\mathbf{w}$, and generally has a single global minimum. To obtain the minimum value of $E\{e^2(n)\}$, taking the partial derivative on the r.h.s. of (3.1.3) with respect to $\mathbf{w}$, and making it equal to zero, the optimal solution, i.e., the Wiener solution for $\mathbf{w}$ is then obtained

$$\mathbf{w}_{opt} = R_x^{-1}\mathbf{r}_{dx} \tag{3.1.4}$$

where $\mathbf{r}_{dx}$ is an L-by-1 cross-correlation vector

$$\mathbf{r}_{dx} = \begin{bmatrix} r_{dx}(0) \\ \vdots \\ r_{dx}(L-1) \end{bmatrix} = \begin{bmatrix} E\{d(n)x(n)\} \\ \vdots \\ E\{d(n)x(n-L+1)\} \end{bmatrix} \tag{3.1.5}$$

and R_x is an L-by-L autocorrelation matrix

$$R_x = E\{\mathbf{x}\mathbf{x}^T\} \tag{3.1.6}$$

Equation (3.1.4) is called the Wiener-Hopf equation.

The Wiener solution is a block-based solution, i.e., it requires that all the input and the desired signal samples are available, since the statistical values are utilized in (3.1.4). Furthermore, the matrix inverse operation is also needed in (3.1.4), which results in a heavy computational complexity if the tap-length L is large. Both properties limit its application in practice.

Next the steepest descent algorithm is introduced, which is designed to establish the Wiener solution $\mathbf{w}_{opt}$ in an iterative way to avoid the matrix inverse operation.

3.1.2 The steepest descent algorithm

In the steepest descent algorithm the filter coefficient vector is updated in an iterative way and it will approach the Wiener solution at steady-state. The steepest descent method is a general scheme that uses the following steps to search for the minimum point of any convex function of a set of parameters [15]:

1. Start with an initial guess of the parameters whose optimum values are to be found for minimizing the function.

2. Find the gradient of the function with respect to these parameters at their present values.

3. Update the parameters by taking a step in the opposite direction of the gradient vector obtained in step 2.

4. Repeat steps 2 and 3 until no further significant change is observed in the parameters.

The steepest descent algorithm to establish the Wiener solution can then be described as

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu \nabla E\{e^2(n)\} \tag{3.1.7}$$

where $\mathbf{w}(n)$ is the adaptive coefficient vector, μ is the step size which controls the convergence rate, and $\nabla E\{e^2(n)\}$ is the gradient of the MSE with respect to $\mathbf{w}(n)$. From (3.1.3) the gradient vector $\nabla E\{e^2(n)\}$ can be formulated as

$$\nabla E\{e^2(n)\} = -E\{e(n)\mathbf{x}(n)\} \tag{3.1.8}$$

Substituting (3.1.2) into (3.1.8) yields

$$\nabla E\{e^2(n)\} = E\{[d(n) - \mathbf{w}^T(n)\mathbf{x}(n)]\mathbf{x}(n)\} \tag{3.1.9}$$

Utilizing (3.1.5) and (3.1.6) equation (3.1.9) becomes

$$\nabla E\{e^2(n)\} = \mathbf{r}_{dx} - R_x\mathbf{w}(n) \tag{3.1.10}$$

Substituting (3.1.10) into (3.1.7) the steepest descent algorithm coefficient update can then be formulated as

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu[\mathbf{r}_{dx} - R_x\mathbf{w}(n)] \tag{3.1.11}$$

where typically $\mathbf{w}(0) = \mathbf{0}$. As can be seen in (3.1.11) the matrix inverse operation is no longer needed in the steepest descent algorithm.

However, the statistical values are still necessary for the update.

3.1.3 The LMS algorithm

In practice, the statistical values used in the steepest decent algorithm are normally unknown. The LMS algorithm is obtained by replacing these statistical values with their instantaneous estimates:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}(n) \tag{3.1.12}$$

where

$$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n) \tag{3.1.13}$$

The LMS algorithm provides a recursive adaptation of the filter coefficient vector with every new arriving sample. The eminent feature of the LMS algorithm is that the adaptation is very simple, as can be seen in (3.1.12). It requires only $2L+1$ multiplications and $2L$ additions. Furthermore, it is a very robust algorithm, since the update in (3.1.12) itself has a time average operation on instantaneous gradient estimates. It has been widely used in many signal processing applications such as in system identification, adaptive noise cancellation and linear prediction.

The step size parameter μ in (3.1.12) plays a very important role for the LMS algorithm. According to the analysis in [15], the convergence in the mean condition for the step size can be formulated as

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{3.1.14}$$

where λ_{max} is the maximum eigenvalue of the autocorrelation matrix R_x. The convergence rate can be denoted by the time constant τ, which

is defined as the maximum value of the time for the absolute values of all the elements of the deviation vector $\mathbf{w}_{opt} - \mathbf{w}(n)$ to reach $1/e$ of their initial amplitudes, and can be formulated as [15]

$$\tau = [-\frac{1}{\ln(1-\mu\lambda_{min})}] \approx \frac{1}{\mu\lambda_{\min}} \tag{3.1.15}$$

where λ_{min} is the minimum eigenvalue of the matrix R_x. It is straightforward to see in (3.1.15) that with the increase of the step size, the convergence time constant τ will decrease, which indicates that the convergence rate will increase.

With the assumptions that both the input and the noise signals are statistically stationary, the data vector $\mathbf{x}(n)$ and the coefficient vector $\mathbf{w}(n)$ are statistically independent, and the step size is sufficiently small, the steady state excess mean square error (EMSE) which is defined as $\lim_{n\to\infty} \xi^2(n) = E\{[e(n) - t(n)]^2\}$ can be formulated as [16]:

$$\xi^2(\infty) \approx \frac{\mu\sigma_v^2 Tr(R_x)}{2} \tag{3.1.16}$$

where σ_v^2 is the variance of the noise signal and $Tr(\cdot)$ denotes the trace of the matrix.

It is clear to see from (3.1.16) that the steady state EMSE is proportional to the step size. Together with equation (3.1.15) the conclusion can be drawn that the step size value provides a tradeoff between the convergence rate and the steady state EMSE. An intuitive way to improve the performance of the LMS algorithm is to make the step size variable rather than fixed, i.e., choose large step size values during the initial convergence of the LMS algorithm, and use small step size values when the system is close to its steady-state, so that both a fast con-

vergence rate and a small steady-state EMSE can be obtained. This results in variable step size LMS (VSSLMS) algorithms, which will be discussed in Chapter 5.

Note that in all the above formulations, the adaptive filter coefficient vector $\mathbf{w}(n)$ is assumed to have the same tap-length as the optimal coefficient vector $\mathbf{w}_{opt}$. However, in certain situations the tap-length of the optimal coefficient vector is unknown or variable, and variable tap-length LMS (VTLMS) algorithms are needed to find a proper choice for the tap-length. The topic of VTLMS algorithms is the focus of Chapter 6.

3.2 BSS: problem formulation

BSS algorithms are designed to recover unobservable source signals from observed mixtures with the assumption that the sources are independent [17] and [18]. Due to various potential applications in communications, speech signal processing and biomedical signal processing, it has received much attention recently. From the perspective of the mixing model, the BSS problem can be divided into two classes: the instantaneous BSS problem, which is normally solved by independent component analysis (ICA) methods, and the convolutive BSS problem, which is more complex and more close to reality.

3.2.1 Instantaneous mixtures of sources

The problem of blind source separation is traditionally approached by observing instantaneous mixtures of sources. Assuming that N source signals $s_i(n)$ are ordered in a vector $\mathbf{s}^T(n) = [s_1(n), ..., s_N(n)]$, upon transmission through a medium these signals are collected by M sen-

sors, from which an observed vector $\mathbf{x}^T(n) = [x_1(n), ..., x_M(n)]$ is obtained. For simplicity, $N = M$ is always assumed in the reminder of this chapter. This assumption is idealistic in as much as there will be many sources in a real acoustic environment, however, it provides a framework for the initial work in the area of reverberation time estimation in occupied rooms described in this thesis. On the basis of linear superposition the vector $\mathbf{x}(n)$ can be formulated as

$$\mathbf{x}(n) = H\mathbf{s}(n) + \mathbf{v}(n) \tag{3.2.1}$$

where H is assumed to be an invertible N-by-N matrix and called the mixing matrix (situations such as when sources arrive from identical directions which cause it to be singular are not considered), and $\mathbf{v}(n) = [v_1(n), ..., v_N(n)]$ is the noise vector. The objective is to recover the original signals $s_i(n)$ given only the observed vector $\mathbf{x}(n)$ and the assumption that all the source signals are independent of each other. The recovered signals can be formulated as

$$\mathbf{s}(n) = H^{-1}\mathbf{x}(n) \tag{3.2.2}$$

However, the inverse matrix H^{-1} is unavailable since the mixing matrix H is unknown, as the term blind suggests. The goal of blind source separation is to find an N-by-N matrix W such that the components of the reconstructed signals

$$\mathbf{y}(n) = W\mathbf{x}(n) \tag{3.2.3}$$

where $\mathbf{y}^T(n) = [y_1(n), ..., y_N(n)]$ are mutually independent, without knowing the mixing matrix H and the probability distribution of the source signals $\mathbf{s}(n)$. Ideally, the matrix W is expected to be the inverse of the mixing matrix H. In practice, the unmixing matrix W should satisfy that

$$WH = AD \tag{3.2.4}$$

where A is a permutation matrix, i.e. all the elements of each column and row are zero except for one element with value unity, and D is a diagonal matrix. The existence of the matrix A and D is because that signals obtained by changing the order of independent source signals or their amplitudes will still be independent, and satisfy the independence assumption.

3.2.2 Convolutive mixtures of sources

Practically, perfectly instantaneous mixtures of sounds are seldom encountered. For example, in the acoustic field the observed signals which are collected by microphones are convolutive mixtures of source signals because of the reverberant environment, in which reflections will make the mixing model more complex. Similar to the previous subsection, supposing that there are N mutually independent source signals $s_i(n)$ and N observed signals $x_i(n)$, the convolutive mixing model can be formulated as

$$x_i(n) = \sum_{j=1}^{N} \sum_{p=0}^{P-1} h_{ij}(p) s_j(n-p) + v_i(n) \qquad i = 1, ..., N \tag{3.2.5}$$

where $h_{ij}(p)$ represents the impulse response from source j to microphone i and P is the length of the impulse response.

Again, the goal of convolutive BSS algorithms is to separate observed signals $x_i(n)$ into N independent signals, without knowing the mixing impulse responses or the probability distribution information of the original source signals $s_i(n)$. This is normally achieved by searching for the unmixing FIR filter $w_{ij}(q)$ with a tap-length of Q, so that the signals obtained as

$$y_i(n) = \sum_{j=1}^{N} \sum_{q=0}^{Q-1} w_{ij}(q) x_j(n-q) \qquad i = 1, ..., N \tag{3.2.6}$$

are mutually independent. It is clear to see from (3.2.5) that when $P = 1$ and $Q = 1$, the convolutive BSS problem will be identical to the instantaneous BSS problem. When $N = 1$, the convolutive BSS problem becomes the blind deconvolution problem [18]. The convolutive BSS problem involves the inverse of the mixing filters. However, only the FIR filters with minimum phase have causal infinite impulse response (IIR) inverse filters. In practice, the unmixing filters are modelled as FIR filters, as shown in (3.2.6), and estimated to make the unmixing signals $y_i(n)$ to be as mutually independent as possible.

3.3 Instantaneous BSS algorithms

In the standard blind source separation problem, the mixtures are assumed to be instantaneous. Herault and Jutten seem to be the first to have addressed the problem of source separation [19]. Comon [20] formulated the problem of separating instantaneous linear mixtures and clearly defined the term independent component analysis (ICA), which is an efficient way to solve the instantaneous BSS problem.

One class of ICA algorithms is based on information theory and

utilizes the non-Gaussianity of the source signals [20] [21] [22] [23]. Essentially, all these algorithms are designed to approximately measure the independence between the separated signals by higher order statistics. Also there are many methods that have been proposed which utilize second order statistics [24] [25] [26] [27]. Methods proposed in [24] [25] [26] utilize the time structure of the mixture signals, and are designed to simultaneously diagonalize the correlation matrix of observed signals at one or several time delays, while the method proposed in [27] is to simultaneously diagonalize the time-varying covariance matrices. An overview for these approaches can be seen in [28] [29] and the text books [17] [18] [30].

Next, typical principles and algorithms for instantaneous BSS will be introduced.

3.3.1 BSS algorithms utilizing non-Gaussianity of the source signals

A good starting point for developing instantaneous BSS algorithms is to utilize the property that the source signals are mutually independent. By assuming the source signals are statistically stationary, and at most one signal is Gaussian distributed, the independence of the reconstructed signals can be measured by the Kullback-Leibler divergence between their joint distribution and the product of their marginal distributions. This Kullback-Leibler divergence is also named as the mutual information of the separated signals. Since the signals are assumed to be statistically stationary, for convenience of formulation, all the signals $x_i(n)$, $s_i(n)$ and $y_i(n)$ will be replaced with random variables x_i, s_i and y_i in this subsection. The mutual information of the separated

signals can then be formulated as

$$I(y_1, ..., y_N) = \int_{-\infty}^{\infty} p_y(\mathbf{u}) \cdot \log \frac{p_y(\mathbf{u})}{\prod p_{y_i}(u_i)} d\mathbf{u} \tag{3.3.1}$$

where $\int$ denotes the integral operation, $p_y(\cdot)$ is the joint probability density function (PDF) of the separated signals, and $p_{y_i}(\cdot)$ is the marginal PDF of the ith separated signal y_i. The mutual information formulated in (3.3.1) is always positive and zero only if the separated signals y_i are mutually independent, thus it is a good cost function for the BSS algorithms.

The mutual information can also be written in terms of the entropy of the separated signals

$$I(y_1, ..., y_N) = \sum_{i=1}^{N} H(y_i) - H(\mathbf{y}) \tag{3.3.2}$$

where $H(\cdot)$ denotes the entropy of random variables

$$\begin{aligned} H(\mathbf{y}) &= -\int_{-\infty}^{\infty} p_y(\mathbf{u}) \log p_y(\mathbf{u}) d\mathbf{u} \\ H(\mathbf{y}) &= -\int_{-\infty} p_y(\mathbf{u}) \log p_y(\mathbf{u}) d\mathbf{u} \end{aligned} \tag{3.3.3}$$

and

$$H(y_i) = -\int_{-\infty}^{\infty} p_{y_i}(u) \log p_{y_i}(u) du \tag{3.3.4}$$

As can be seen in (3.2.3), the random vector $\mathbf{y}$ is a linear transformation of the observed vector $\mathbf{x}$. Based on information theory the following relationship exists [17]:

$$H(\mathbf{y}) = H(\mathbf{x}) + \log |W| \tag{3.3.5}$$

where $|\cdot|$ indicates the determinant of a matrix. Substituting (3.3.5)

into (3.3.2) yields

$$I(y_1, ..., y_N) = \sum_{i=1}^{N} H(y_i) - H(\mathbf{x}) - \log|W| \tag{3.3.6}$$

The solution of the BSS algorithm then becomes to find an unmixing matrix W to minimize the mutual information of the separated signals. With such an unmixing matrix W, the original source signals can be reconstructed up to an arbitrary scale and possible permutation of indices.

Another approach for ICA is the maximum likelihood estimation method. According to the mixing model formulated in (3.2.1), the joint PDF of the observed vector $\mathbf{x}$ can be formulated as [17]

$$p_x(\mathbf{x}) = |H|^{-1} \prod_{i=1}^{N} p_{s_i}(s_i) \tag{3.3.7}$$

where $p_{s_i}(\cdot)$ is the marginal PDF of the ith source signal s_i. The PDF of the observed vector $\mathbf{x}$ can also be written as

$$p_x(\mathbf{x}) = |H|^{-1} \prod_{i=1}^{N} p_{s_i}(\mathbf{h}_i^T \mathbf{x}) \tag{3.3.8}$$

where $\mathbf{h}_i$ is the ith column of the matrix $(H^{-1})^T$. Since H^{-1} is unavailable in practice, it is replaced with an estimate W. The log-likelihood function of the observed samples $\mathbf{x}(1), ..., \mathbf{x}(S)$ with respect to the estimated unmixing matrix W can then be obtained [17]

$$\log L(W) = \sum_{n=1}^{S} \sum_{i=1}^{N} \log p_{s_i}(\mathbf{w}_i^T \mathbf{x}(n)) + S \log|W| \tag{3.3.9}$$

where $\mathbf{w}_i$ is the ith column of the matrix W^T. By approximating the

sample average in (3.3.9) with the statistical average, equation (3.3.9) can be approximately rewritten as

$$\frac{1}{S}\log L(W) = \sum_{i=1}^{N} E\{\log p_{s_i}(\mathbf{w}_i^T\mathbf{x})\} + \log|W| \tag{3.3.10}$$

Utilizing the definition of the entropy in (3.3.4), and assuming the PDF of the separated signals are very close to that of the source signals, equation (3.3.10) can also be written as

$$\frac{1}{S}\log L(W) \approx -\sum_{i=1}^{N} H(y_i) + \log|W| \tag{3.3.11}$$

It is clear to see that equation (3.3.11) is equivalent to equation (3.3.6) except the global sign and the additive constant given by $H(\mathbf{x})$. Thus the criterion of maximizing the likelihood of the observed signals is the same as that of minimizing the mutual information of the separated signals.

Both criteria formulated in (3.3.6) and (3.3.11) can not be used directly in practice, since the PDF of the source signals are unknown. To deriving practical BSS algorithms, different approximations of the PDF and different optimization approaches have been utilized [17] [18] [20] [21] [22].

The first class of approaches is to approximate the PDF of the signals by higher order cumulants. Since the term $H(\mathbf{x})$ is a constant, and independent of the unmixing matrix W, minimizing the mutual information formulated in (3.3.6) is equivalent to minimize the term $\sum_{i=1}^{N} H(y_i) - \log|W|$. A reasonable constraint for the unmixing matrix is that the determinant of the unmixing matrix $|W|$ is a constant, since the solution of the BSS itself has an amplitude ambiguity. The criterion

of minimizing the mutual information then becomes to minimize the sum of entropies of the separated signals $\sum_{i=1}^{N} H(y_i)$. The entropy is also a measurement of the Gaussianity, or non-Gaussianity of the signal, since a Gaussian variable has the largest entropy among all random variables with equal variance. A more convenient measurement of the non-Gaussianity is the negentropy, which is defined as:

$$J(y) = H(y_{gauss}) - H(y) \tag{3.3.12}$$

where y_{gauss} is a Gaussian random variable with the same variance as y. In practice, the negentropy can be approximated by higher order statistics

$$J(y) = \frac{1}{12}E\{y^3\}^2 + \frac{1}{48}[kurt(y)]^2 \tag{3.3.13}$$

where $kurt(\cdot)$ denotes the kurtosis of the variable, which can be formulated as

$$kurt(y) = E\{y^4\} - 3[E\{y^2\}]^2 \tag{3.3.14}$$

When the random variables have approximately symmetric distributions, which is very common in practice, the first term in (3.3.13) will be very small, thus the criterion of minimizing the mutual information is approximated as maximizing the negentropy, which is again approximated as maximizing the kurtosis of the separated signals. By replacing the kurtosis with its time average estimate, gradient based algorithms can be utilized to search for the unmixing matrix W [17]. The criterion discussed above is essentially equivalent to maximizing the non-Gaussianity of the separated signals, which can also be explained based upon the central limit theorem, i.e., sums of non-Gaussian random variables are closer to Gaussian than the original ones. Therefore, a linear

combination of the observed mixture variables will be maximally non-Gaussian if it equals one of the independent components. This is why the condition that at most one signal is Gaussian is necessary for higher order statistics based BSS algorithms.

The second class of approaches is to approximate the unknown PDF of source signals by some nonlinear functions. Substituting (3.2.3) into (3.3.6) and differentiating (3.3.6) with respect to W yields [17]

$$\frac{\partial I(y_1,...,y_N)}{\partial W} = E\{\mathbf{g}(W\mathbf{x})\mathbf{x}^T\} + [W^T]^{-1} \tag{3.3.15}$$

where $\mathbf{g}(\mathbf{y}) = [g_1(y_1),...,g_N(y_N)]$, and the function $g_i(\cdot)$ is a nonlinear function which is approximated as

$$g_i(\cdot) \approx (\log p_{s_i})' = \frac{p'_{s_i}}{p_{s_i}} \tag{3.3.16}$$

By replacing the statistic value in (3.3.16) with its instantaneous estimate the update of the unmixing matrix W becomes

$$\Delta W \propto \mathbf{g}(W\mathbf{x})\mathbf{x}^T + [W^T]^{-1} \tag{3.3.17}$$

This algorithm was firstly derived in [21], although by using the infor-max principle, which in essence is equivalent to the ML approach [23].

It is shown in [22] that the unmixing matrix parameter space has a Riemannian metric structure, and the natural gradient works more efficiently as compared with the gradient approach formulated in (3.3.17). The natural gradient can be formulated in terms of the gradient as

$$\frac{\partial J}{\partial W_{nat}} = \frac{\partial J}{\partial W} W^T W \tag{3.3.18}$$

where J is the cost function, W is the parameter matrix to be found, $\frac{\partial J}{\partial W_{nat}}$ is the natural gradient, and $\frac{\partial J}{\partial W}$ is the gradient. Utilizing (3.3.17) and (3.3.15), the natural gradient algorithm is then obtained

$$\Delta W \propto (I + \mathbf{g}(\mathbf{y})\mathbf{y}^T)W \tag{3.3.19}$$

As shown in [17], a good choice of the nonlinear function $g(\cdot)$ for super-Gaussian source signals is

$$g(y) = -\tanh(y) \tag{3.3.20}$$

and for sub-Gaussian source signals is

$$g(y) = \tanh(y) - y \tag{3.3.21}$$

or

$$g(y) = y^3 \tag{3.3.22}$$

It is clear to see in the above discussion that the key step for the instantaneous BSS algorithms utilizing non-Gaussianity of the source signals is to approximate the PDF of the source signals by higher order statistics or nonlinear functions. It has been shown that both approaches can result in good separation performance, although these approximations are likely to be fairly inaccurate [17].

3.3.2 BSS algorithms utilizing time structure of the source signals

In the previous subsection, BSS algorithms utilizing the non-Gaussianity of the source signals have been discussed. These algorithms can not solve the BSS problem with Gaussian distributed source signals. It will

be shown in this subsection that second order statistics obtained from the time structure of the observed signals can also be used to perform BSS, without the non-Gaussianity condition.

One class of second order statistics based algorithms is to eliminate the temporal cross-correlation functions of the separated signals as much as possible, i.e., to simultaneously diagonalize the autocorrelation matrix of the observed signals at one or several time delays. According to the linear mixing model formulated in (3.2.1), the autocorrelation matrix of the observed signals with a time lag τ can be formulated as

$$\mathbf{R}_x(\tau) = E\{\mathbf{x}(t)\mathbf{x}^T(t-\tau)\} \tag{3.3.23}$$

With the assumption of independence, the autocorrelation matrix of the source signals is a diagonal matrix

$$E\{\mathbf{s}(t)\mathbf{s}^T(t-\tau)\} = \Lambda_s \tag{3.3.24}$$

Defining a modified version of the autocorrelation matrix of the observed matrix as

$$\overline{\mathbf{R}}_x(\tau) = \frac{1}{2}(\mathbf{R}_x(\tau) + \mathbf{R}_x^T(\tau)) \tag{3.3.25}$$

then

$$\overline{\mathbf{R}}_x(\tau) = W^T\Lambda_s W \tag{3.3.26}$$

and the rows of the unmixing matrix W can be obtained as the eigenvectors of the matrix $\overline{\mathbf{R}}_x(\tau)$. In practice, the autocorrelation matrix is obtained by sample averaging. This algorithm is termed as the algorithm for multiple unknown signals extraction (AMUSE) algorithm [25]. A similar approach can be seen in [24]. It is clear to see that if there

is no linear time correlations for the observed signals, for example, if $E\{\mathbf{x}(t)\mathbf{x}^T(t-\tau) = 0\}$ for any lag $\tau \neq 0$, the AMUSE method will fail. This method can be extended to several time lags, i.e., to simultaneously diagonalize all the corresponding lagged covariance matrices, which results in the second-order blind identification (SOBI) algorithm [26].

Another class of approaches is to utilize the nonstationary of the observed signals. In these approaches, different estimated autocorrelation matrices at different times can be obtained, due to the nonstationary of the source signals, and the unmixing matrix W can be obtained by joint diagonalizing all these autocorrelation matrices in an adaptive way [27].

In general, estimation of higher order statistics is more sensitive to noise and outliers than that of second order statistics, and the resulting cost functions often suffer from undesired local minima especially in the adaptive algorithms. Second order statistics have the advantage that they can be estimated more reliably using less computational power than higher order statistics. It is clear to see that in practice, the choice of the criterion for ICA depends on the property of the source signals. The criteria discussed in this section can also be utilized in convolutive BSS algorithms, as will be shown in the next section.

3.4 Convolutive BSS algorithms

The convolutive BSS problem is different from the instantaneous one due to the difference of the mixing models, as can be seen in Section 3.2. However, with the same assumption that the source signals are mutually independent, similar criteria to those utilized in the instantaneous BSS algorithms can be used in the convolutive case. Some convolutive

BSS algorithms are performed in the time domain [31] [32] [33], while others are designed in the frequency domain [9] [34] [35] [36] [37] [38]. The frequency domain methods are more attractive for applications where the unknown unmixing filter tap-length is long, since in contrast to time domain approaches, where a large number of coefficients of the unmixing filters have to be estimated, frequency domain approaches simplify the convolutive BSS problem into the instantaneous BSS problem at each frequency bin. The number of frequency bins is equal to the DFT length which is chosen from prior knowledge of the unknown mixing filter impulse response sequence length and the nonstationarity of the input signals [39]. However, the permutation problem occurs at each frequency bin, and must be solved to recover the separated signals correctly from the frequency domain. In this section, both typical time domain and frequency domain convolutive BSS algorithms are described, and the permutation problem will be discussed in the next section.

3.4.1 Time domain convolutive BSS algorithms

Many time domain convolutive BSS algorithms are obtained by extending the instantaneous BSS algorithms into the convolutive case [31] [32] [33], and a summary work for convolutive BSS algorithms can be found in the book [18].

As an example, a NG convolutive BSS algorithm is proposed in [31], where the unmixing matrix is directly updated as

$$\Delta W_q(n) \propto W_q(n) - \mathbf{g}(\mathbf{y}(n-Q))\mathbf{u}^T(n-q), \quad q = 0, ..., Q-1 \quad (3.4.1)$$

where

$$\mathbf{u}(n) = \sum_{i=0}^{Q-1} W_{Q-i}^{T}(n)\mathbf{y}(n-i) \tag{3.4.2}$$

and W_q is an N-by-N matrix composed of the elements $w_{ij}(q)$, $\mathbf{g}(\cdot)$ is the nonlinear function the same as that in (3.3.15). Note no pre-whitening is required by this algorithm. It is clear to see that the update of the unmixing matrix in (3.4.1) is very similar to that in (3.3.19), and will be the same if $Q = 1$.

Although time domain convolutive BSS algorithms have been shown to have good separation performance when the mixing filter length is short, which is close to the instantaneous cases, the performance will degrade and the computational complexity will increase if the mixing filter length is long, as is the case for room impulse responses [39]. As will be shown in the next subsection, a more efficient way for convolutive BSS is to transform the problem into the frequency domain.

3.4.2 Frequency domain convolutive BSS algorithms

Many investigators transform the problem into the frequency domain to solve an instantaneous BSS problem for every frequency bin simultaneously. The advantages of the frequency domain methods mainly include mathematical simplicity, reduction of the computational complexity, and quick convergence property.

By using a T-point discrete Fourier transform (DFT) (for discussions of the selection of T see [39]), the time domain mixing signal $x_i(n)$ can be written in the frequency domain as

$$x_i(\omega, n) = \sum_{\tau=0}^{T-1} x_i(n+\tau)w(\tau)e^{-j2\pi\omega\tau} \tag{3.4.3}$$

where $w(\tau)$ denotes a window function, $j = \sqrt{-1}$ and ω is sampled over the frequency range $\omega = 0, \frac{1}{T}2\pi, ..., \frac{T-1}{T}2\pi$. With the convolutive mixing model formulated in (3.2.5), as assumed by many workers in the field [34] [36] [37] [38] [9], a compact form for the frequency domain convolutive mixing model can be obtained

$$\mathbf{x}(\omega, n) = H(\omega)\mathbf{s}(\omega, n) \tag{3.4.4}$$

where $\mathbf{x}(\omega, n) = [x_1(\omega, n), ..., x_N(\omega, n)]^T$, $\mathbf{s}(\omega, n) = [s_1(\omega, n), ..., s_N(\omega, n)]^T$ and $H(\omega)$ is assumed to be an N-by-N invertible matrix composed of $h_{ij}(\omega)$, which is the frequency representation for the mixing impulse response $h_{ij}(p)$. It is clear to see that by utilizing a DFT, the time domain convolutive mixing model formulated in (3.2.5) becomes an instantaneous mixing model at each frequency bin, as can be seen in (3.4.4). The frequency domain BSS problem then becomes an instantaneous BSS problem at each frequency bin. By utilizing the instantaneous BSS algorithms discussed in the previous section, an N-by-N frequency domain separating matrix $W(\omega)$ can be found for each frequency bin, so that the frequency domain separated signals

$$\mathbf{y}(\omega, n) = W(\omega)\mathbf{x}(\omega, n) \tag{3.4.5}$$

are mutually independent, where $W(\omega)$ is an N-by-N matrix composed of $w_{ij}(\omega)$, which is the frequency domain representation for the unmixing impulse response $w_{ij}(q)$. The existence of the inverse follows from the assumption that $H(\omega)$ is invertible at each frequency bin. The time domain separated signals $\mathbf{y}(n)$ can then be obtained from $\mathbf{y}(\omega, n)$ by using an inverse DFT (IDFT) operation. By using differ-

ent criteria and approaches discussed in the previous section, different convolutive BSS algorithms can then be obtained, as can be seen in [34] [35] [36] [37] [38] [9].

As an example, a frequency domain convolutive BSS algorithm namely Parra & Spence's method [9] which is used in the BSS stage in the RT estimation framework is introduced in detail. Similar to the criterion of the ICA algorithms that exploit the statistical nonstationarity of the source signals, Parra & Spence's method jointly diagonalizes the autocorrelation matrices at different times for each frequency bin. An advantage of this method is that the uncorrelated noise can be estimated and removed from the separated signals.

Considering the mixing model formulated in (3.2.5), the autocorrelation matrix of the observed signals at one frequency bin can be approximated by the sample mean

$$\overline{R}_x(\omega,n)=\frac{1}{N}\sum_{i=0}^{N-1}\mathbf{x}(\omega,n+iT)\mathbf{x}^T(\omega,n+iT) \tag{3.4.6}$$

With appropriate choice of N, which is proportional to the signal length and inversely proportional to the length of the DFT, as discussed in [40], from the independence assumption the estimated autocorrelation matrix can be written as

$$\overline{R}_x(\omega,n)\approx H(\omega)\Lambda_s(\omega,n)H^T(\omega)+\Lambda_v(\omega,n) \tag{3.4.7}$$

where $\Lambda_s(\omega,n)$ and $\Lambda_v(\omega,n)$ are the time-frequency formulations of the autocorrelation matrices for the source signals and the noise signals, and both are diagonal matrices. Thus the unmixing matrix $W(\omega)$ should satisfy that the estimated autocorrelation matrix of the source signals

which is obtained as

$$\hat{\Lambda}_s(\omega,n) = W(\omega)[\overline{R}_x(\omega,n) - \Lambda_v(\omega,n)]W^T(\omega) \tag{3.4.8}$$

can be a diagonal matrix. Dividing the observed signals into K sections, K estimated autocorrelation matrices can be obtained from (3.4.6). By utilizing the nonstationary of the observed signals, the unmixing matrix is then updated to simultaneously diagonalize these K autocorrelation matrices, or equivalently, to simultaneously minimize the off-diagnonal elements of the K matrices obtained from (3.4.8). The cost function can then be formulated as

$$J = \sum_{\omega=0}^{T-1}\sum_{k=1}^{K} \|E(\omega,k)\|^2 \tag{3.4.9}$$

where $\|\cdot\|^2$ denotes the Euclidean norm and

$$\|E(\omega,k)\|^2 = W(\omega)[\overline{R}_x(\omega,k) - \Lambda_v(\omega,k)]W^T(\omega) - \Lambda_s(\omega,k) \tag{3.4.10}$$

The least squares estimates of the unmixing matrix and the autocorrelation matrices of the source signals and noise signals can be obtained as

$$\hat{W}, \hat{\Lambda}_n, \hat{\Lambda}_s = \arg \min_{W,\Lambda_n,\Lambda_s} J \tag{3.4.11}$$

The gradients of the cost function are

$$\frac{\partial J}{\partial W(\omega)} = 2\sum_{k=1}^{K} E(\omega,k)W(\omega)[\overline{R}_x(\omega,t) - \Lambda_v(\omega,k)] \tag{3.4.12}$$

$$\frac{\partial J}{\partial \Lambda_s(\omega,k)} = -diag[E(\omega,k)] \tag{3.4.13}$$

$$\frac{\partial J}{\partial \Lambda_n(\omega,k)} = -diag[W^T(\omega)E(\omega,k)W(\omega)] \tag{3.4.14}$$

where $diag(\cdot)$ denotes the diagonalization operator which zeros the off-diagonal elements of the matrix. The optimal unmixing matrix $W(\omega)$ and the noise autocorrelation matrix $\Lambda_v(\omega,k)$ can then be obtained by a gradient descent algorithm using the gradients formulated in (3.4.12) and (3.4.14), and the autocorrelation matrix of the source signals can be obtained by setting the gradient in (3.4.13) to zero. A more detailed introduction for this approach can be seen in [9].

Although it has been shown that frequency domain methods are more efficient and have a better convergence property, the permutation problem in the frequency domain is more serious as compared with that in the time domain, since the blind estimated unmixing matrix at one frequency bin can at best be obtained up to a scale and permutation. Therefore, at each frequency bin the separated signal $s_i(\omega)$ can be recovered at an arbitrary output channel. Consequently, the recovered source signal is not necessarily a consistent estimate of the real source over all frequencies. So it is necessary to solve the permutation problem and align the unmixing matrix to an appropriate order, so that the source signals can be recovered correctly. This problem will be discussed in the next section. The scale ambiguity problem at each frequency bin is mitigated simply by normalizing the unmixing matrix at each frequency bin [10].

3.5 Solving the permutation problem

Typical methods for solving the permutation problem are introduced in this section. These methods can be divided into four classes:

1. Exploiting unmixing matrix spectral continuity.

2. Exploiting the similarity of the signal envelope structure at different frequency bins from the same source signal.

3. Utilizing geometrical constraints, i.e., exploiting the difference of the positions of the source signals.

4. Combined methods, which use the advantages of different approaches, whilst avoid their disadvantages.

3.5.1 Exploiting unmixing matrix spectral continuity

Parra & Spence [9] suggest to solve the permutation problem by imposing a smoothness constraint on the unmixing filters, i.e., a constraint on the filter length $Q < T$, where T is the FFT window length. This finite length constraint in the time domain forces the solution to be smooth or continuous in the frequency domain, thus promoting convergence to a global minimum.

Smaragdis [34] [35] worked wholly with this problem in the frequency domain by using ICA algorithms such as the natural gradient algorithm for each frequency bin. To solve the permutation problem, he proposed an adaptive scheme to exploit frequency coupling between neighboring frequency bins. The adaptation of unmixing matrics is formulated as follows:

$$\Delta W_f = \Delta W_f + k\Delta W_{f-1}, \quad 0 < k < 1 \tag{3.5.1}$$

where W_f is the unmixing matrix W at frequency bin f.

Obviously, both methods have an implicit assumption of spectral continuity, i.e., the neighboring unmixing matrix are similar to each

other.

3.5.2 Exploiting signal envelope structure

Murata et al. [38] solve the permutation problem by exploiting the similarity of the signal envelope structure at different frequency bins from the same source signal. This method can be formulated as follows:

1.Sort ω in order of the weakness of correlation between the separated signals $y_i(\omega, n)$, i.e., ω_1 is chosen as the frequency bin in which the separated signals have the weakest correlation.

2.For ω_1, fix the order of separated signals.

3.For ω_k find the permutation of the order of separated signals which maximizes the correlation between the envelope of the separated signals $y_i(\omega_k, n)$ and the aggregated envelope of the separated signals from $y_i(\omega_1, n)$ through $y_i(\omega_{k-1}, n)$.

4.Assign the appropriate permutation to all the frequency bins.

It is clear to see that this approach implicitly assumes that at different frequency bins, the envelope structure from the same signal is similar.

3.5.3 Utilizing geometrical constraints

All BSS methods make no assumption about the positions of the sources in the 3-dimensional space. As shown in [41] and [40], the source signals generally originate from different spatial locations in practice, and this geometrical information can be utilized to solve the permutation problem. This approach is motivated by the beamforming technique, which estimates the direction of arrival (DOA) of signals in order to steer the beam of an array of sensors to focus on a specific source [42].

A beamformer consists of an array of sensors in a particular configuration. The output of each sensor is properly filtered and the filtered outputs of all the sensors are added up. Typically, a beamformer linearly combines the spatially sampled waveform from each sensor in the same way as an FIR filter which linearly combines temporally sampled data. In the beamforming technique, the beamformer response is defined as the amplitude and phase presented to a plain wave as a function of location and frequency. Consider the signal is a plain wave with DOA θ and frequency ω, and for convenience let the phase be zero at the first sensor, the sensor array response vector can be generally expressed as

$$\mathbf{d}(\theta,\omega) = [1, e^{j\omega\tau_2(\theta)}, ..., e^{j\omega\tau_N(\theta)}]^T \tag{3.5.2}$$

where τ_i is the time delay of the source signal at the ith sensor, $j = \sqrt{-1}$. The beamformer response can then be formulated as

$$r(\theta,\omega) = \mathbf{w}^T\mathbf{d}(\theta,\omega) \tag{3.5.3}$$

where $\mathbf{w}$ is the beamformer filter vector. It is straightforward to see in (3.5.3) that the angle between $\mathbf{w}$ and $\mathbf{d}(\theta,\omega)$ determines the response $r(\theta,\omega)$. The ability to discriminate sources at different locations and frequency is obtained by utilizing the difference of angles of their array response vectors.

It is clear to see that the model used in the beamformer is very similar to that of BSS, in which the row vector of the unmixing matrix can be deemed as a beamformer, and one source from one direction is extracted. The equivalence between frequency-domain blind source separation and frequency-domain adaptive beamforming for convolu-

tive mixtures is discussed in detail in [43]. By constraining the beamformer response to be a constant, the permutation of the unmixing matrix at different frequency bins can be removed [41] and [40]. Similar approaches can be seen in [44] [45] and [46].

3.5.4 Combined methods

A combined approach for solving the permutation problem is proposed in [47], which uses both the signal envelope structure and the geometrical information of source signals. A more robust and precise method for the permutation problem is obtained by utilizing the good properties of both single approaches, whilst avoiding their disadvantages. The advantage of the geometrical information based method is the robustness since a misalignment at one frequency bin does not affect other frequencies. However, DOA cannot be well estimated at some frequencies, especially at low frequencies where the phase difference caused by the sensor spacing is very small, and also at high frequencies where spatial aliasing might occur. In essence, the DOA approach is not precise since the evaluation is based on an approximation of the mixing system. The correlation approach is not robust since a misalignment at one frequency bin may cause consecutive misalignments. The correlation approach is precise as long as signals are well separated by ICA since the measurement is based on separated signals.

The combined approach integrates the two approaches to solve the permutation problem in the two following steps:

1) Fix the permutations at some frequencies where the confidence of the DOA approach is sufficiently high;

2) Decide the permutations for the remaining frequencies based on

correlations of signal envelope structure without changing the permutations fixed by the DOA approach.

As compared with each single approach, the combined method is more robust and precise [47].

3.6 Chapter Summary

The background introduction for the adaptive techniques, i.e., the LMS algorithm and BSS algorithms, is provided in this chapter. The purpose was to give a broad discussion as details are available in the cited works. Both techniques will be utilized in the RT estimation framework, as will be described in the next section. The introduction also provides a basis for the further research on adaptive techniques, i.e., the VSSLMS algorithms in Chapter 5, the VTLMS algorithms in Chapter 6 and a variable tap-length NG algorithm in Chapter 7.

Chapter 4

A COMBINED BSS AND ANC SCHEME WITH POTENTIAL APPLICATION IN BLIND ACOUSTIC PARAMETER EXTRACTION

For most existing RT estimation methods, including the maximum likelihood estimation (MLE) based method introduced in Chapter 2, the condition of low noise level is necessary to obtain accurate RT estimates. With the increase of the noise level, the RT estimates will generally be biased, or even wrong, since the noise will contaminate the fine structure of the received excitation signal. To improve the accuracy of the RT estimates, a preprocessing is introduced in this chapter, in which the blind source separation (BSS) technique and adaptive noise cancellation (ANC) scheme, based upon the least mean square (LMS) algorithm are combined to reduce the unknown noise level from the passively received speech signal. As a demonstration this preprocessing will be used together with the MLE based method to estimate the

RT of a synthetic noise room. Simulation results show that the proposed new approach can improve the accuracy of the RT estimation in a simulated high noise environment. The potential application of the proposed approach for realistic acoustic environments is also discussed, which motivates the need for further development of more sophisticated frequency domain BSS algorithms.

This chapter is organized as follows: the novel framework which utilizes the proposed preprocessing together with the MLE based RT estimation method is introduced in Section 4.1. The BSS stage of the framework is discussed in Section 4.2. The ANC stage of the framework is described in Section 4.3. The MLE based RT estimation stage is formulated in Section 4.4. Simulations are given in Section 4.5, in which the proposed approach is utilized to extract the RT in a synthetic high noise room. The potential application of the proposed approach in a real acoustic environment is discussed in Section 4.6. Section 4.7 provides the conclusion.

4.1 Introduction of the proposed RT estimation framework

As have been introduced in Chapter 1 and Chapter 2, many methods have been proposed to estimate the RT [4] [6] [5]; 'blind' methods which utilize the passively received speech signals are particularly attractive in certain environments as good controlled excitation signals are unnecessary [7] [8]. In [7] an artificial neural network (ANN) is trained to extract the RT from passively received speech utterances. In [8], an exponentially damped Gaussian white noise model is used to describe the reverberation tail of the received speech signal, and an MLE method is then performed on segments of the speech signal to extract the RT. As

shown by the authors, both of these algorithms provide reliable RT estimates in a noise free environment. When estimating RT in high noise environments, however, the results of these methods will be degraded and generally biased, as can be seen in [3] and the later simulations. Therefore both methods are limited by the noise level.

To make the RT estimation methods more robust and accurate, an intuitive way is to remove the unknown noise signal from the received speech signal as much as possible before the RT estimation. A powerful tool for extracting some noise interference signal from a mixture of signals is the convolutive BSS method [9]. Naturally, given two spatially distinct observations, BSS can attempt to separate the mixed signals to yield two independent signals. One of these two signals mainly consists of the excitation speech signal plus residue of the noise and the other signal contains mostly the noise signal [9]. The estimated noise signal can then serve as a reference signal within an ANC, which is then used to remove the noise component contained in the received speech signal. The different stages of this framework are shown in Fig. 4.1. The signal $s_1(n)$, which is assumed to be the noise signal in this work, is assumed statistically independent of the excitation speech signal $s_2(n)$. The passively received signals $x_1(n)$ and $x_2(n)$ are modelled as convolutive mixtures of $s_1(n)$ and $s_2(n)$. The room impulse response $h_{ij}(n)$ is the impulse response from source j to microphone i. BSS is used firstly to obtain the estimated excitation speech signal $\hat{s}_2(n)$ and the estimated noise signal $\hat{s}_1(n)$. The estimated noise signal $\hat{s}_1(n)$ then serves as the reference signal for the ANC to remove the noise component from $x_1(n)$. The output of the ANC $\hat{y}_{12}(n)$ is an estimation of the noise free reverberant speech signal $y_{12}(n)$. As compared with $x_1(n)$, it crucially

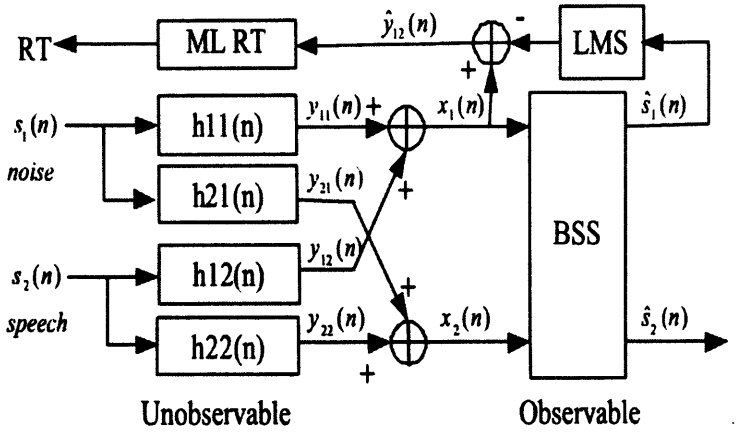

Figure 4.1. Proposed blind RT estimation framework

retains the reverberant structure of the speech signal and has a low level of noise, therefore it is more suitable to estimate the RT of the occupied room. As will be shown by the later simulations, the proposed method can improve the accuracy of the RT estimation in a simulated high noise environment. Note that due to the symmetrical structure of the proposed approach, the signal $s_1(n)$ can also be deemed as an excitation speech signal, the signal $s_2(n)$ can then be deemed as a noise signal, and a similar approach can be performed to extract the RT. Likewise, $x_2(n)$ can also be used as a basis for estimating RT at the expense of additional computational complexity. This aspect is however left as future work.

From Fig. 4.1 it is clear to see that the key stage of the proposed method is the BSS stage. The performance of the whole framework depends on the performance of BSS. If a good estimation of the noise signal can be obtained, the noise contained in the mixture signal can be removed by the ANC, and the output of the ANC can be a good signal to estimate the RT. In the rest of this chapter, different stages of the proposed method will be described, and a detailed discussion and

simulation study of its performance will be given.

4.2 BSS stage

As shown by Fig. 4.1, the goal of BSS is to extract the estimated noise signal $\hat{s}_1(n)$ from the received mixture signals $x_1(n)$ and $x_2(n)$. By assuming that the room environment is time invariant, the received mixtures $x_1(n)$ and $x_2(n)$ can be modelled as weighted sums of convolutions of the source signals $s_1(n)$ and $s_2(n)$. The equation that describes this convolved mixing process is:

$$x_i(n) = \sum_{j=1}^{2} \sum_{p=0}^{P-1} s_j(n-p) h_{ij}(p) + v_i(n), \quad i = 1, 2 \tag{4.2.1}$$

where $s_j(n)$ is the source signal from a source j, $x_i(n)$ is the received signal by a microphone i, $h_{ij}(p)$ is the P-point response from source j to microphone i, and $v_i(n)$ is additive white noise, which is assumed to be zero, as indicated in Fig. 4.1. Using a T-point windowed discrete Fourier transformation (DFT), the time domain signal $x_i(n)$ can be converted into the time-frequency domain signal $x_i(\omega, n)$ where ω is a frequency index and n is a time index. For each frequency bin the following equation is obtained

$$\mathbf{x}(\omega, n) = H(\omega)\mathbf{s}(\omega, n) + \mathbf{v}(\omega, n) \tag{4.2.2}$$

where $\mathbf{s}(\omega, n) = [s_1(\omega, n), s_2(\omega, n)]^T$, $\mathbf{x}(\omega, n) = [x_1(\omega, n), x_2(\omega, n)]^T$ and $\mathbf{v}(\omega, n) = [v_1(\omega, n), v_2(\omega, n)]^T$ are the time-frequency representations of the source signals, the observed signals and the noise signals, $H(\omega)$ is a 2-by-2 matrix composed of $h_{ij}(\omega)$, which is the frequency representation

for the mixing impulse response $h_{ij}(p)$, $(\cdot)^T$ denotes vector transpose. The separation can be completed by a 2-by-2 unmixing matrix $W(\omega)$ of a frequency bin ω

$$\hat{\mathbf{s}}(\omega, n) = W(\omega)\mathbf{x}(\omega, n) \tag{4.2.3}$$

where $\hat{\mathbf{s}}(\omega, n) = [\hat{s}_1(\omega, n), \hat{s}_2(\omega, n)]^T$ is the time-frequency representation of the estimated source signals and $W(\omega)$ is the frequency representation of the unmixing matrix. $W(\omega)$ is determined so that $\hat{s}_1(\omega, n)$ and $\hat{s}_2(\omega, n)$ become mutually independent.

As a demonstration of the proposed approach, the frequency domain convolutive BSS method which exploits the nonstationary of the observed signals is utilized [9]. For each frequency bin, exploiting the statistical nonstationarity of the speech signal, the unmixing matrix $W(\omega)$ can be found by jointly diagonalizing K autocorrelation matrices of the observed signals at K different times [9]. This approach has also been introduced in the previous chapter. The separated signals $\hat{s}_1(n)$ and $\hat{s}_2(n)$ can then be obtained from (4.2.3) after applying an inverse DFT (IDFT). The scale ambiguity problem as previously discussed is addressed by matrix normalization [10] and the permutation problem is mitigated by using the unmixing filter tap-length constraint, as discussed in [9]. One advantage of this approach is that it incorporates uncorrelated noise, although in practice the noise may degrade its performance.

The performance of BSS can be evaluated by checking the separated noise signal $\hat{s}_1(n)$ or the excitation signal $\hat{s}_2(n)$. According to the above description, the frequency domain mathematical representation of the

separated noise signal can be formulated as

$$\hat{s}_1(\omega, n) = c_{11}(\omega)s_1(\omega, n) + c_{12}(\omega)s_2(\omega, n) \tag{4.2.4}$$

where $c_{11}(\omega)$ and $c_{12}(\omega)$ are the frequency-domain representations of the combined system responses:

$$c_{11}(\omega) = h_{11}(\omega)w_{11}(\omega) + h_{21}(\omega)w_{12}(\omega) \tag{4.2.5}$$

and

$$c_{12}(\omega) = h_{12}(\omega)w_{11}(\omega) + h_{22}(\omega)w_{12}(\omega) \tag{4.2.6}$$

The performance of BSS can be classified into three possible cases:

1. A perfect performance, which is obtained if the separated noise signal can be approximately deemed as a scaled or delayed version of the original noise signal. In this case, the z-domain representation of the combined response filter c_{11} can be formulated as $c_{11}(z) = Cz^{-\Delta}$ where C is a scalar and Δ is an integer to denote the delay, and the combined system response c_{12} is close to zero.

2. A good performance, which is obtained if the separated noise signal is approximately a filtered version of the source noise signal. In this case, the filter c_{11} is an unknown filter, and the filter c_{12} is approximately zero.

3. A normal performance, which is obtained if the separated noise signal contains both components of the original noise signal and the excitation speech signal. In this case, both filters c_{11} and c_{12} are two unknown filters. In this situation the performance of the adaptive noise canceller may also be degraded but this aspect is not considered in the

thesis.

If the performance of BSS is perfect, the estimation of the structure of the room impulse responses can be obtained from the inverse of the unmixing filters, and the room RT can then be estimated directly from the room impulse responses. However, in real applications, the performance of most existing BSS algorithms are between case 2 and case 3, and the inverse of the unmixing matrix filters will be seriously biased from the room impulse responses. This is why an extra ANC stage is needed in the proposed framework.

4.3 ANC stage

After the BSS stage the estimated noise signal $\hat{s}_1(n)$ is obtained, which is highly correlated with the noise signal $s_1(n)$. This signal is then used as a reference signal in the ANC stage to remove the noise component from the received signal $x_1(n)$. A introduction of the LMS algorithm can be seen in the previous chapter. Since the target signal $y_{12}(n)$ which is to be recovered is a highly nonstationary speech signal, a modified LMS algorithm namely the sum method [48] is combined with the proportional adaptation [49] in this ANC stage to obtain both a fast convergence rate and a small steady state excess mean square error (EMSE). The update of this new approach, which is named as the proportional sum method, can be summarized as follows:

$$e(n) = x_1(n) - \hat{\mathbf{s}}_1^T(n)\mathbf{w}(n) \tag{4.3.1}$$

$$l(n) = max\{|w_1(n)|, ..., |w_L(n)|\} \tag{4.3.2}$$

$$l'(n) = max\{\delta, l(n)\} \tag{4.3.3}$$

$$g_k(n) = max\{\rho l'(n), |w_k(n)|\} \quad k = 1, ..., L \tag{4.3.4}$$

$$\bar{g}(n) = \frac{1}{L}\sum_{i=1}^{L} g_i(n) \tag{4.3.5}$$

$$w_k(n+1) = w_k(n) + \frac{\mu e(n) g_k(n) \hat{s}_{1,k}(n)}{\bar{g}(n) L[\hat{\sigma}_e^2(n) + \hat{\sigma}_s^2(n)]} \quad k = 1, ..., L \tag{4.3.6}$$

where $e(n)$ is the output error of the adaptive filter, $\hat{\mathbf{s}}_1(n)$ is the input vector with a tap-length of L, $\hat{\mathbf{s}}_1(n) = [\hat{s}_1(n), ..., \hat{s}_1(n-L+1)]$ and $\hat{s}_{1,k}(n)$ is its kth element, $\mathbf{w}(n)$ is the weight vector of the adaptive filter and $w_k(n)$ is its kth element, $|\cdot|$ denotes the absolute value operation, $l(n)$ is the maximum absolute value of the elements of the adaptive filter vector, $l'(n)$ is a slight modification of $l(n)$ which avoids the situation when all the absolute values of the elements of the adaptive filter vector are small by utilizing the constant δ, ρ is another constant which is used to avoid the situation that at some iterations $w_k(n)$ may equal to zero, $g_k(n)$ is the gain distributor of the kth element of the adaptive filter, $\bar{g}(n)$ is approximately the average value of the absolute values of the elements of the adaptive filter vector, $\hat{\sigma}_e^2(n)$ and $\hat{\sigma}_s^2(n)$ are estimations of the temporal error energy and the temporal input energy, which are obtained by first order smoothing filters:

$$\hat{\sigma}_e^2(n) = 0.99\hat{\sigma}_e^2(n-1) + (1-0.99)e^2(n) \tag{4.3.7}$$

and

$$\hat{\sigma}_s^2(n) = 0.99\hat{\sigma}_s^2(n-1) + (1-0.99)\hat{s}_1^2(n) \tag{4.3.8}$$

where $\hat{\sigma}_s^2(0) = 0$ and $\hat{\sigma}_e^2(0) = 0$. The choice of 0.99 is related to the window length of such estimates and is approximately $\frac{1}{1-0.99} = 100$. Moreover, the term $1 - 0.99$ ensures unbiased estimates.

The proportional adaptive filter is suitable for applications where the energy of the adaptive filter elements are distributed unevenly over L taps [49]. Intuitively, gain distributors are proportional to the magnitude of the current impulse response sample estimates. Tap weights that are currently being estimated as far from zero get significantly more update energy than those currently being estimated as close to zero, which speeds up the convergence rate with a similar EMSE as compared with the normalized LMS (NLMS) algorithm. More details and discussions of the proportional adaptive filter can be found in [49].

The adaptation of the weight vector in (4.3.6) is based on the sum method in [48]. As explained by the author, the adaptation in (4.3.6) is adjusted by the input and output error variance automatically, which reduces the influence brought by the fluctuation of the input and the target signals.

If the BSS stage performs well, the output signal of the ANC $\hat{y}_{12}(n)$ should be a good estimation of the noise free reverberant speech signal $y_{12}(n)$. By denoting the steady state adaptive filter vector as $\mathbf{w}_s$ and its frequency domain representation as $w_s(\omega)$, the time-frequency domain representation of $\hat{y}_{12}(n)$ can be formulated as follows:

$$\begin{aligned}\hat{y}_{12}(\omega,n) &= x_1(\omega,n) - w_s(\omega)\hat{s}_1(\omega,n) \\ &= g_1(\omega)s_1(\omega,n) + g_2(\omega)s_2(\omega,n) \end{aligned} \tag{4.3.9}$$

where $g_1(\omega)$ and $g_2(\omega)$ are combined system responses:

$$g_1(\omega) = h_{11}(\omega) - w_s(\omega)c_{11}(\omega) \tag{4.3.10}$$

and

$$g_2(\omega) = h_{12}(\omega) - w_s(\omega)c_{12}(\omega) \tag{4.3.11}$$

where $c_{11}(\omega)$ and $c_{12}(\omega)$ are formulated in (4.2.5) and (4.2.6). With the three kinds of performance of BSS which are discussed in the previous section, three kinds of performance of the ANC are obtained, based on the output signal $\hat{y}_{12}(n)$:

1. If the BSS stage has a perfect performance, according to the discussion in the previous section the following equation is obtained

$$\hat{y}_{12}(\omega, n) = [h_{11}(\omega) - w_s(\omega)Ce^{-j\Delta\omega}]s_1(\omega, n) + h_{12}(\omega)s_2(\omega, n) \tag{4.3.12}$$

It is clear to see that if the adaptive filter of the ANC converges to the value of $w_s(\omega) = \frac{h_{11}(\omega)e^{j\Delta\omega}}{C}$, a noise free reverberant speech signal $\hat{y}_{12}(n) = y_{12}(n)$ can be obtained at the output of the ANC.

2. If the BSS stage has a good performance, the following equation can be obtained

$$\hat{y}_{12}(\omega, n) = [h_{11}(\omega) - w_s(\omega)c_{11}(\omega)]s_1(\omega, n) + h_{12}(\omega)s_2(\omega, n) \tag{4.3.13}$$

To remove the noise component from $\hat{y}_{12}(n)$, the first term of the right hand side of (4.3.13) should be equal to zero, which results in $w_s(\omega) = h_{11}(\omega)/c_{11}(\omega)$. It requires that the combined system response $c_{11}(\omega)$ has an inverse. Thus if the inverse of the combined system response is not realizable, the ANC can not remove the noise component completely. According to the simulation experience, in most cases the ANC stage can partially remove the noise component from the received mixture signal, and improve the accuracy of the RT estimates.

3. If the BSS stage has a normal performance, the output signal of the ANC will contain both components of the noise signal and the speech signal. The reverberant structure contained in $\hat{y}_{12}$ will be damaged. In practice, if the performance of BSS is poor, the noise contained in the mixture signal can not be removed, and an extra noise component will be introduced.

If both the BSS stage and the ANC stage perform well, the output signal of the ANC stage can then be used for acoustic parameter extraction. In the next section the MLE based RT estimation method will be introduced.

4.4 MLE based RT estimation method

In this method, the RT of an occupied room is extracted from a passively received speech signal [8]. At first, the passively received speech signal is divided into several overlapped segments with the same length. Each segment can be deemed as an observed vector. This observed vector is modelled as an exponentially damped Gaussian random sequence, i.e., it is modelled as an element-by-element product of two vectors, one is a vector with an exponentially damped structure, and the other is composed of independent identical distributed (i.i.d.) Gaussian random samples. Note that the exponentially damped vector also models the envelope of the speech segment. The MLE approach is applied to the observed vector to extract the decay rate of its envelope. The RT can then be easily obtained from the decay rate, according to its definition. An estimation of RT can be extracted from one segment, and a series of RT estimates can be obtained from the whole passively received speech signal. The most likely RT of the room can then be identified from

these estimates.

In the MLE based RT estimation stage, the fine structure of the reverberant tail of the output signal $\hat{y}_{12}(n)$ is overlap segmented by a window with a width of N. At each segment an observed vector $\mathbf{y}_N$ is obtained. The mathematical formulation for the exponentially damped Gaussian random sequence model used on $\mathbf{y}_N$ is as follows [8]:

$$\mathbf{y}_N(i) = \mathbf{x}_N(i)\mathbf{a}_N(i), i = 1...N \tag{4.4.1}$$

where $\mathbf{x}_N$ is a vector whose elements are drawn from a random white Gaussian sequence $x(n) \sim (0, \sigma^2)$ and $\mathbf{a}_N$ is an exponentially damped sequence whose elements are determined by $\mathbf{a}_N(i) = a^i, i = 1...N$ where $a = 1/\exp(-\tau)$, τ is a constant which describes the damping rate. It is easy to see that τ actually describes the damping rate of sequence $\mathbf{a}_N(i)$, which is used to model the envelope of the reverberant speech signal. According to the definition the RT can be obtained from this decay rate:

$$T_{60} = 6.91\tau \tag{4.4.2}$$

By using an MLE approach, both the parameters a and σ can be obtained according to the model formulated in (4.4.1) [14]. With the estimate of parameter a, the decay parameter τ and the RT can also be calculated. From each segment an estimate of RT can be obtained, and a series of estimates of RT can be obtained with the total output signal $\hat{y}_{12}(n)$. These estimates can then be used to identify the most likely RT of the room by using an order-statistic filter [8]. A simple and intuitive way to identify the RT from a series estimations is to choose the peak of a histogram of the RT estimations. Detailed introduction

of the MLE based RT estimation method can be seen in Chapter 2.

In the next section the proposed framework is utilized to extract the RT of a simulated high noise room. The RT estimates obtained from the proposed approach will be compared with the original MLE approach to show its advantage.

4.5 Simulation

In this section the performance of the proposed approach is examined. To confirm the discussion in previous sections, three simulations are performed based on different performance of the BSS stage. The flow chart of the simulations is shown in Fig. 4.1. All these simulations are based on the same environment: the simulated room and its impulse responses h_{ij} between source j and microphone i are simulated by a simplistic image room model which generate only positive impulse response coefficients [50]. The room size is set to be 10*10*5 meter^3 and the reflection coefficient is set to be 0.7 in rough correspondence with the actual room. The RT of this room measured by Schroeder's method [6] is 0.27s. The excitation speech signal and the noise signal are two anechoic 40 seconds male speech signals with a sampling frequency of 8kHz, and scaled to have a unit variance over the whole observation. The first 10s of these two signals can be seen in Fig. 4.2. The position of these two sources are set to be [1m 3m 1.5m] and [3.5m 2m 1.5m]. The positions of the two microphones are set to be [2.45m 4.5m 1.5m] and [2.55m 4.5m 1.5m] respectively. The impulse responses h_{11}, h_{12}, h_{21}, h_{22} are shown in Fig. 4.3. The setup of the simulation can be seen in Fig. 4.4. The selection of the room geometry is a typical example of many examples tried in the related simulation studies.

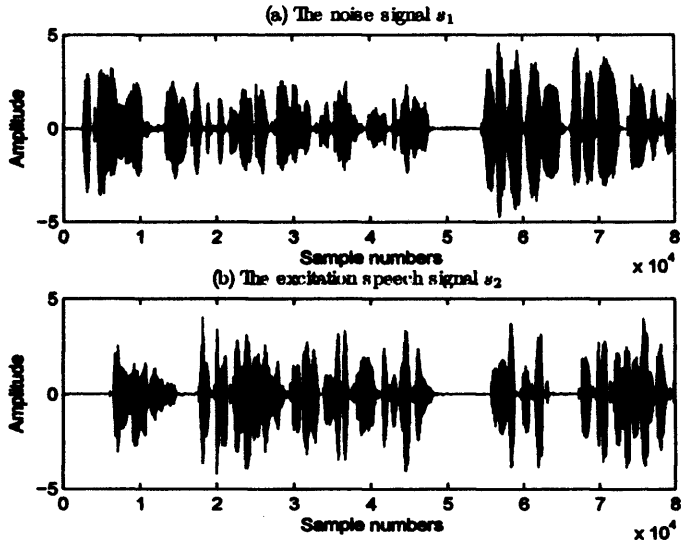

Figure 4.2. The excitation speech signal and the noise signal

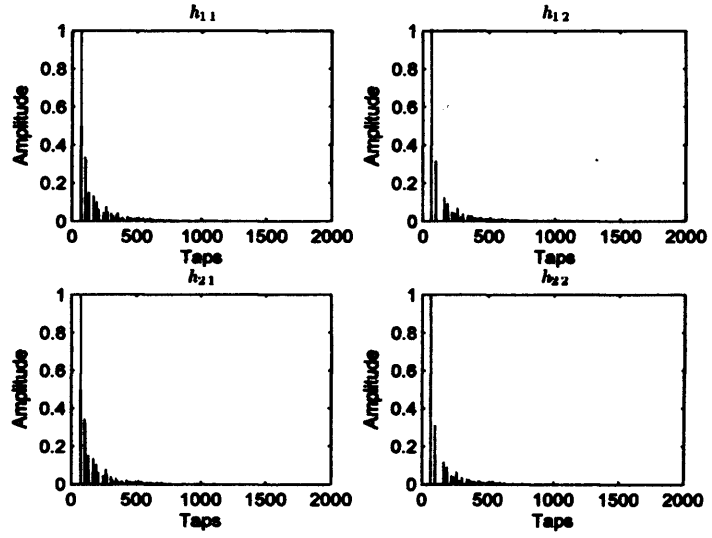

Figure 4.3. Simulated room impulse responses

This example only suggests the potential applicability of the proposed approach for occupied room RT estimation. An extensive evaluation is left for future work, once the component processing schemes have been optimized. The focus of the remainder of this thesis is to investigate improvements for the ANC stage.

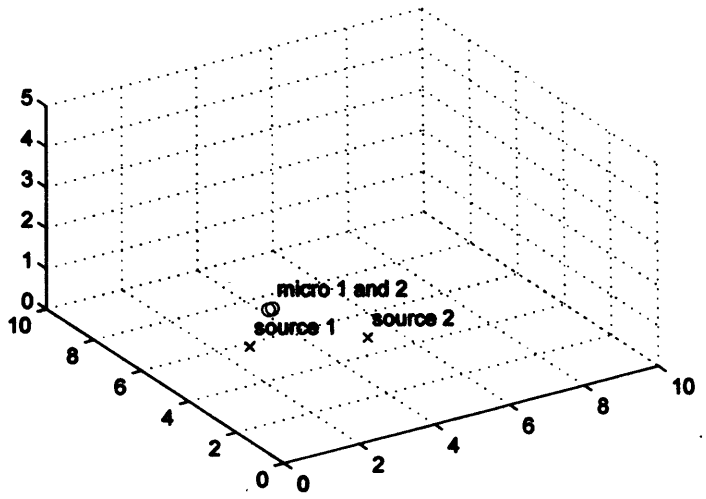

Figure 4.4. Simulated room (unit in meter)

The parameter setting for the BSS algorithm is as follows: the mixture signals are divided into $K = 5$ sections, so that 5 autocorrelation matrices of the mixture signals at each frequency bin are obtained. The DFT length is set to $T = 2048$. The unmixing filter tap-length is set to $Q = 512$, which is much less than T, to reduce the permutation ambiguity [9]. The step size of the update of the frequency domain unmixing matrix is set to unity. The parameter setting for the ANC stage is as follows: the tap-length of the adaptive filter coefficient vector is set to 500. The step size μ is set to 0.005. The parameter δ is set to 0.001. The parameter ρ is set to 0.01. The smoothing parameter β is set to 0.99. The window width which is used to obtain the observed vector in the MLE based RT estimation method is set to $1,200$. All these parameters have been chosen empirically to yield the best performance. The online method which is introduced in Chapter 2 is used to calculate the RT.

For each simulation, the performance of ANC combined with BSS

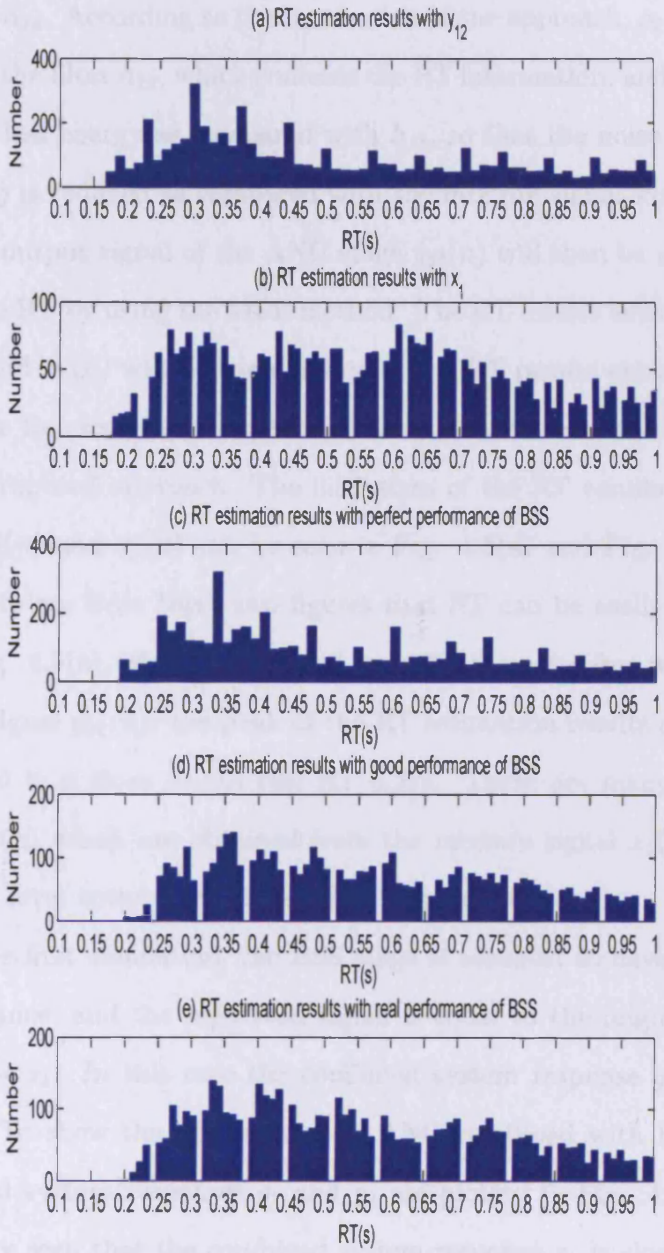

Figure 4.5. The histogram of the RT estimation results with different signals

is shown by comparing the combined system responses g_1 and g_2 which are formulated in (4.3.10) and (4.3.11) with the room impulse responses h_{11} and h_{12}. According to the motivation of the approach, g_2 should be close to the filter h_{12}, which contains the RT information, and g_1 should contain less energy as compared with h_{11}, so that the noise contained in $\hat{y}_{12}(n)$ is reduced as compared with the mixture signal $x_1(n)$.

The output signal of the ANC stage $\hat{y}_{12}(n)$ will then be used to extract the RT by using the MLE method. The RT results extracted from $\hat{y}_{12}(n)$ and $x_1(n)$ will be compared with the RT results extracted from the noise free reverberant speech signal $y_{12}(n)$, to show the advantage of the proposed approach. The histogram of the RT results extracted from $y_{12}(n)$ and $x_1(n)$ can be seen in Fig. 4.5(a) and Fig. 4.5(b). It is clear to see from these two figures that RT can be easily identified from Fig. 4.5(a), which is obtained by using the noise free reverberant speech signal $y_{12}(n)$: the peak of the RT estimation results appears at 0.3s, and it is close to the real RT 0.27s. There are many peaks in Fig. 4.5(b) which are obtained from the mixture signal $x_1(n)$ due to the high level noise, thus RT is difficult to be identified.

In the first simulation, the BSS stage is assumed to have a perfect performance, and the separated signal is equal to the original signal, i.e., $\hat{s}_1 = s_1$. In this case the combined system response g_2 is equal to h_{12}. To show the performance of ANC combined with BSS, both combined system responses g_1 and g_2 are plotted in Fig. 4.6. It can be clearly seen that the combined system response g_1 is close to zero, which indicates the output signal $\hat{y}_{12}(n)$ is very close to the noise free reverberant speech signal $y_{12}(n)$, according to (4.3.12).

The RT extracted from signal $\hat{y}_{12}$ is shown in Fig. 4.5(c). It is clear

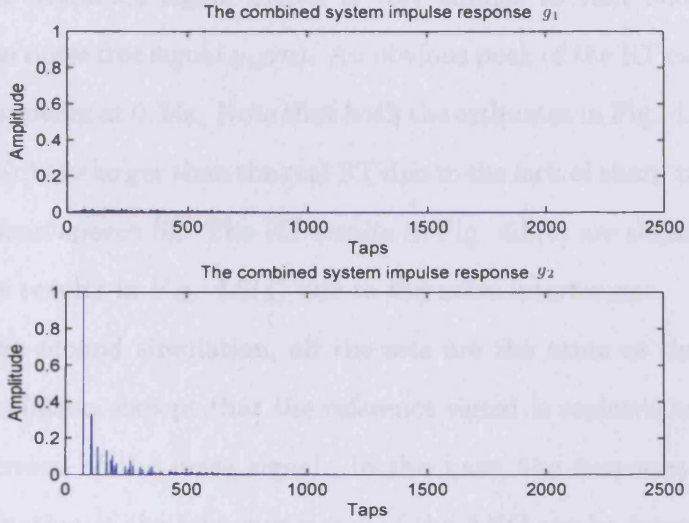

Figure 4.6. Combined system responses with a perfect performance of BSS

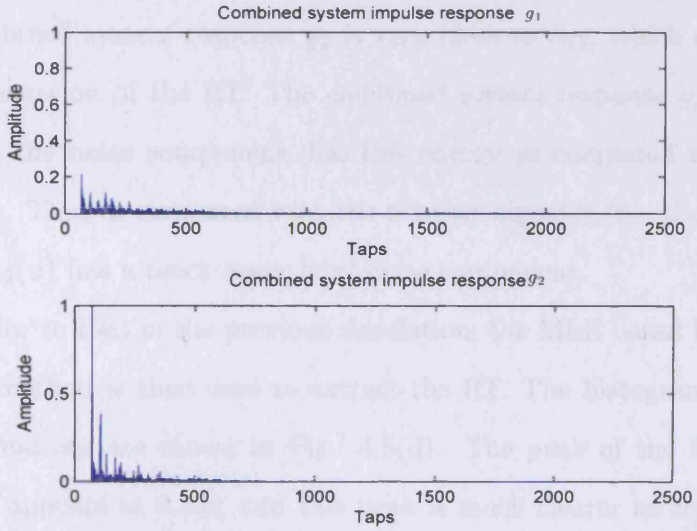

Figure 4.7. Combined system responses with a good performance of BSS

to see in Fig. 4.5 that the histogram of the RT estimations obtained with the extracted signal $\hat{y}_{12}(n)$ is very similar to that obtained by using the noise free signal $y_{12}(n)$. An obvious peak of the RT estimation results appears at 0.34s. Note that both the estimates in Fig. 4.5(a) and Fig. 4.5(c) are larger than the real RT due to the lack of sharp transients in the clean speech [8]. The RT results in Fig. 4.5(c) are slightly larger than the results in Fig. 4.5(a) due to the noise interference.

In the second simulation, all the sets are the same as that of the first simulation, except that the reference signal is replaced with a filtered version of the noise signal. In this case, the frequency domain representation of the reference signal of the ANC can be formulated as

$$\hat{s}_1(\omega, n) = c_{11}(\omega) s_1(\omega, n) \tag{4.5.1}$$

where $c_{11}(\omega)$ is formulated in (4.2.5). The combined system responses g_1 and g_2 are shown in Fig. 4.7. From Fig. 4.7 it is clear to see that the combined system response g_2 is very close to h_{12}, which contains the information of the RT. The combined system response g_1, which contains the noise component, has less energy as compared with the filter h_{11}. Thus as compared with the mixture signal $x_1(n)$, the output signal $\hat{y}_{12}(n)$ has a much lower level noise component.

Similar to that of the previous simulation, the MLE based RT estimation method is then used to extract the RT. The histogram of the RT estimations are shown in Fig. 4.5(d). The peak of the RT estimations appears at 0.35s, and this peak is much clearer as compared with that in Fig. 4.5(b), which indicates that the result obtained from $\hat{y}_{12}(n)$ is better than the result obtained from $x_1(n)$.

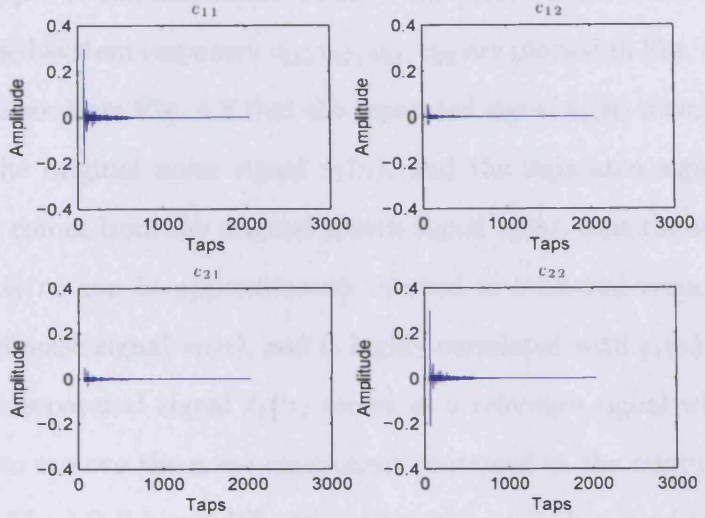

Figure 4.8. Combined system responses c_{11}, c_{12}, c_{21}, c_{22} of BSS

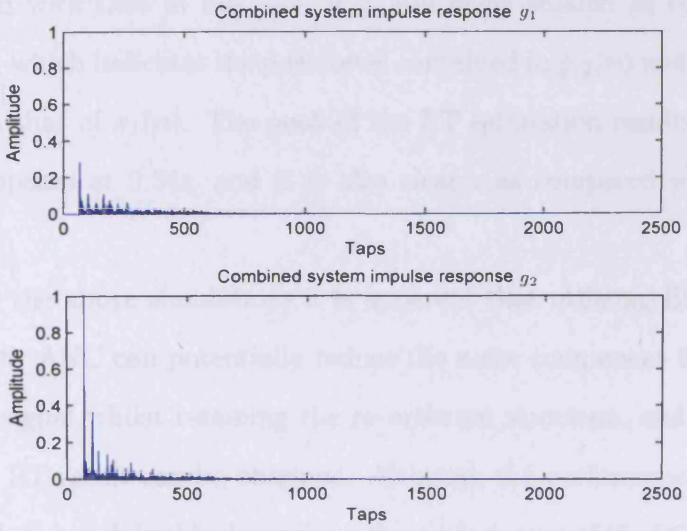

Figure 4.9. Combined system responses with a real performance of BSS

In the last simulation, the reference input of the ANC stage is the real output of the BSS stage. To show the performance of the BSS, the combined system responses c_{11}, c_{12}, c_{21}, c_{22} are plotted in Fig. 4.8. It is clear to see from Fig. 4.8 that the separated signal $\hat{s}_1(n)$ mainly comes from the original noise signal $s_1(n)$, and the separated signal $\hat{s}_2(n)$ mainly comes from the original speech signal $s_2(n)$, thus the separated signal $\hat{s}_1(n)$ can be approximately deemed as a filtered version of the original noise signal $s_1(n)$, and is highly correlated with $s_1(n)$.

The separated signal $\hat{s}_1(n)$ serves as a reference signal within the ANC, to remove the noise component contained in the mixture signal $x_1(n)$. The MLE based RT estimation method is then used to extract the RT estimates from the output signal $\hat{y}_{12}(n)$. The combined system responses g_1 and g_2 are shown in Fig. 4.9, and the histogram of the RT estimations are shown in Fig. 4.5(e). From Fig. 4.9 it is clear to see that although the combined filter g_1 contains more energy as compared with that in Fig. 4.7, it is still much smaller as compared with h_{11}, which indicates the noise level contained in $\hat{y}_{12}(n)$ is still much less than that of $x_1(n)$. The peak of the RT estimation results in Fig. 4.5(e) appears at 0.34s, and it is also clearer as compared with Fig. 4.5(b).

From the above simulations it is apparent that utilizing BSS combined with ANC can potentially reduce the noise component from the mixture signal whilst retaining the reverberant structure, and a more accurate RT result can be obtained. Although the performance of the proposed approach highly depends on the performance of the BSS stage, as shown by the simulations above, nonetheless, the accuracy of the RT estimation is improved, and reliable RT can be extracted by using this

method within a simulated highly noisy room, something that has not previously been possible. Typically, from much simulation experience, the adaptive schemes must improve the SNR at the input to the MLE RT estimation to 15dB, hence the focus of this thesis is now to enhance the ANC to move towards achieving this goal. Future work will also be required to enhance the frequency domain BSS algorithms.

4.6 Discussion

It is clear to see in the above simulations that the key stage of the proposed approach is the BSS stage. Although, as shown in the simulation in the previous section, BSS is successfully used in a simple simulated room impulse response model, the application of BSS and ANC in acoustic parameter extraction is still limited in practice, mainly because of the performance of BSS for a real environment. Generally, there are three fundamental limitations of the BSS stage:

1. The permutation problem in the frequency domain BSS algorithms, as has been introduced in Chapter 2. Since the blind estimated unmixing matrix at one frequency bin can at best be obtained up to a scale and permutation, at each frequency bin the separated signal can be recovered at an arbitrary output channel. Consequently, the recovered source signal is not necessarily a consistent estimate of the real source over all frequencies. So it is necessary to solve the permutation problem and align the unmixing matrix to an appropriate order, so that the source signals can be recovered correctly. Many methods have been proposed to solve the permutation problem. In [9] the authors suggest to solve the permutation problem by imposing a smoothness constraint on the unmixing filters, i.e., exploiting unmixing matrix spectral con-

tinuity. This method is used in the simulations in the thesis. The authors in [38] solve the permutation problem by exploiting the similarity of the signal envelope structure at different frequency bins from the same source signal. Geometrical constraints are used in [40] and [46] to solve the permutation problem. A combined approach which utilizes both the similarity of the signal envelope structure and geometrical constraints is proposed in [47]. It has been shown that the combined approach is more robust and precise as compared with other methods, and can solve the permutation problem quite well. It must be pointed out that solving the permutation problem can improve the performance of frequency domain BSS algorithms only when the mixture signals have been well separated. Thus the separation process at each frequency bin is still the key problem for BSS.

2. The choice of the FFT frame size T. This parameter provides a trade-off between maintaining the independence assumption which is related with the sample number at each frequency bin, and covering the whole reverberation in frequency domain BSS [39]. On one hand, it is constrained that $T > P$ where P is the tap-length of the room impulse response, more strictly, $T > 2P$, so that a linear convolution can be approximated by a circular convolution. On the other hand, if T is too large, the sample number at each frequency bin may be too small, and the independence assumption will collapse. Thus it is important to choose a proper value of the parameter T, as explained in [39].

3. As discussed in [39], the frequency domain BSS system can be understood as two sets of adaptive beamformers (ABFs). As shown in the simulation in [39], although BSS can remove the reverberant jammer sound to some extent, it mainly removes the sound from the

jammer direction, i.e., the unmixing matrix $W(\omega)$ mainly removes the direct sound of the jammer signal, and the other reverberant components which arrive from different directions cannot be separated completely. For the room impulse response model used in this chapter, the energy contained in the reverberant tails is quite small as compared with that of the direct signal, and a good performance of BSS is obtained. For real room impulses, however, the performance of current frequency domain BSS algorithms degrade.

Due to the fundamental limitations of current frequency domain BSS algorithms, more research is needed to make the proposed approach work in more realistic acoustic environments.

4.7 Conclusion

In this chapter, a new preprocessing for room acoustic parameter extraction from high noise rooms by utilizing passively received speech signals is provided. Simulation results show that the noise is reduced greatly by the proposed framework from the reverberant speech signal and the performance of this framework is good in a simulated high noise room environment. The potential application of the proposed approach in practice is also discussed. Due to the motivation of the framework, BSS and ANC can be potentially used together in many acoustic parameter estimation methods as a preprocessing. This framework provides a new way to overcome the noise disturbance in RT estimation. However, to make the proposed approach work in real acoustic parameter extraction, more research is needed, especially on the convolutive BSS algorithms.

Chapter 5

VARIABLE STEP SIZE LMS ALGORITHMS

The LMS algorithm has been extensively used in many applications as a consequence of its simplicity and robustness [15] and [16]. A detailed introduction of the LMS algorithm can be found in Chapter 3. In the application of the LMS algorithm, a key parameter is the step size. As is well known, if the step size is large, the convergence rate of the LMS algorithm will be rapid, but the steady-state mean square error (MSE) will increase. On the other hand, if the step size is small, the steady-state MSE will be small, but the convergence rate will be slow. Thus the step size provides a tradeoff between the convergence rate and the steady-state MSE of the LMS algorithm. An intuitive way to improve the performance of the LMS algorithm is to make the step size variable rather than fixed, i.e., choose large step size values during the initial convergence of the LMS algorithm, and use small step size values when the system is close to its steady-state, which results in variable step size LMS (VSSLMS) algorithms. By utilizing such an approach, both a fast convergence rate and a small steady-state MSE can be obtained.

Many VSSLMS algorithms have been proposed during recent years [51], [52], [53], [54], [55], [56], [57] and [58]. Although these methods

perform well under certain conditions, noise can degrade their performance. In this chapter, two new VSSLMS algorithms are proposed to enhance the convergence rate of the LMS algorithm in a high level statistically stationary or nonstationary noise environment, signal-to-noise ratio (SNR) approximately 0dB, supported by theoretical steady-state performance analysis. As will be shown in the simulations, both proposed algorithms have improved performance as compared with existing VSSLMS algorithms.

This chapter is organized as follows: a concise overview of existing VSSLMS algorithms is provided in Section 5.1. A theoretically optimal VSSLMS algorithm is introduced in Section 5.2. Since for most existing VSSLMS algorithms their performance can be influenced by noise interference, two new VSSLMS algorithms are proposed in Section 5.3 and Section 5.4 for high level noise conditions. One is designed for applications in which the noise is statistically stationary and the other is designed to be robust to statistically nonstationary noise interference. Steady-state performance analysis for both algorithms is provided. Simulations support the theoretical analysis. Finally, conclusions are provided in Section 5.5.

5.1 An overview of VSSLMS algorithms

For convenience of description, the LMS algorithm is formulated firstly within the context of a system identification model. In this case, the zero mean desired signal $d(n)$ is a filtered version of the input signal $x(n)$ corrupted by the uncorrelated noise signal $t(n)$. The mathematical

formulation is as follows:

$$d(n) = \mathbf{x}^T(n)\mathbf{w}_{opt} + t(n) \tag{5.1.1}$$

where $\mathbf{w}_{opt}$ is the unknown optimal filter, $\mathbf{x}(n)$ is the adaptive filter input vector, n denotes the discrete time index and $(\cdot)^T$ denotes the vector transpose operator. The error output of the adaptive filter $e(n)$ is the difference between the desired signal and the output of the adaptive filter:

$$e(n) = d(n) - \mathbf{x}^T(n)\mathbf{w}(n) \tag{5.1.2}$$

where $\mathbf{w}(n)$ is the vector of the adaptive filter weights. The update equation of the LMS algorithm is given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}(n) \tag{5.1.3}$$

where μ is the step size. The excess error is defined as

$$\xi(n) = e(n) - t(n) \tag{5.1.4}$$

and the deviation between the optimal and adaptive filter weight vectors is defined as

$$\mathbf{v}(n) = \mathbf{w}_{opt} - \mathbf{w}(n) \tag{5.1.5}$$

Substituting (5.1.1), (5.1.2), (5.1.3), and (5.1.4) into (5.1.5) the relationship between the excess error and the deviation vector can be formulated as

$$\xi(n) = \mathbf{v}^T(n)\mathbf{x}(n) \tag{5.1.6}$$

In the VSSLMS algorithms, the step size μ is replaced with a vari-

Table 5.1. A summary of the step size updates of certain existing VSSLMS algorithms

Name of the method	Update of the step size
[51] Karni's method 1989	$\mu_{max} = \frac{1}{(L+1)\sigma_x^2}$ $\mu(n) = \mu_{max}(1 - e^{-\alpha\|e(n)\mathbf{x}(n)\|^2})$
[52] Kwong's method 1992	$\mu(n) = \beta\mu(n-1) + \gamma e^2(n)$
[53] Mathews' method 1993	$\mu(n) = \mu(n-1) + \rho e(n)e(n-1)\mathbf{x}^T(n)\mathbf{x}(n-1)$
[54] Aboulnasr's method 1997	$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1)$ $\mu(n) = \alpha\mu(n-1) + \gamma p^2(n)$
[55] Pazaitis' method 1999	$p(n) = \beta p(n-1) + (1-\beta)e^2(n)$ $f(n) = \beta f(n-1) + (1-\beta)e^4(n)$ $C(n) = f(n) - 3p^2(n)$ $\mu(n) = \mu_{max}(1 - e^{-\alpha C(n)})$
[56] Mader's method 2000	$\mu(n) = \frac{E\{\xi^2(n)\}}{\|\mathbf{x}(n)\|^2 E\{e^2(n)\}}$
[57] Ang's method 2001	$\overline{\mathbf{g}}(n) = \beta\overline{\mathbf{g}}(n-1) + e(n-1)\mathbf{x}(n-1)$ $\mu(n) = \mu(n-1) + \gamma e(n)\mathbf{x}^T(n)\overline{\mathbf{g}}(n)$
[58] Shin's method 2004	$\overline{\mathbf{g}}_n(n) = \beta\overline{\mathbf{g}}_n(n-1) + (1-\beta)\frac{\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}e(n)$ $c = \frac{\sigma_i^2}{L\sigma_x^2}$ $\mu(n) = \frac{\mu_{max}\|\overline{\mathbf{g}}_n(n)\|^2}{\|\mathbf{x}(n)\|^2(c+\|\overline{\mathbf{g}}_n(n)\|^2)}$

able parameter $\mu(n)$, and the coefficient vector $\mathbf{w}(n)$ is updated as

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n)e(n)\mathbf{x}(n) \tag{5.1.7}$$

Many VSSLMS algorithms have been proposed to improve the performance of the LMS algorithm by using large step sizes at the early stages of the adaptive process and small step sizes after the system approaches convergence. Typical methods can be found in [51], [52], [53], [54], [55], [56], [57] and [58]. Based on the formulation of the LMS algorithm, the mathematical formulations of the updates of the step size $\mu(n)$ in these algorithms are summarized in Table 5.1.

Note that the input and noise signals are assumed to be statistically stationary in all the formulations in Table 5.1, where σ_x^2 is the variance of the input signal, L is the tap-length of the adaptive filter vector, σ_t^2 is the variance of the noise signal, $\|\cdot\|^2$ denotes the squared Euclidean norm operation, α, β, γ and ρ are positive constants, and guidelines for the choice of these parameters can be found in the corresponding literature, as shown in Table 5.1. The number such as 1989 in the name of the method is the year of publication in the literature.

As can be seen in Table 5.1, the step size in [51] is controlled by the squared Euclidean norm of the instantaneous gradient vector $e(n)\mathbf{x}(n)$. The step size in [52] is controlled by the squared instantaneous error. This method is improved to be robust to uncorrelated noise in [54] by using the squared autocorrelation of errors at adjacent times. The step size in [53] is controlled by the inner product between adjacent gradient vectors. This method is improved in [57], where a smoothing operation is utilized on one gradient vector to reduce the influence of noise interference. The step size in [55] is controlled by the fourth-order cumulant of the instantaneous error.

A theoretically optimal VSSLMS algorithm which can obtain the largest decrease of the mean square deviation (MSD) $E\{\|\mathbf{v}(n)\|^2\}$ at each iteration is proposed in [56], in which the step size is suggested to be proportional to the ratio between the excess mean square error (EMSE) value and the MSE value. The method proposed in [58] can be deemed as a practical version of the algorithm proposed in [56] by using some assumptions and approximations, as will be shown in the next section.

It is clear to see in Table 5.1 that these methods can be divided

into two classes: the methods proposed in [52], [54] and [55] utilize the property that the squared value or forth-order cumulant of the output error is large initially and small at steady-state, while the methods proposed in [51], [53], [57] and [58] are gradient-based VSSLMS algorithms which utilize two properties of the gradient vector:

1. The norm of the gradient vector will be large initially and converge to a small value, potentially zero, at steady-state.

2. The polarity of the gradient vector will generally be consistent during the early stage of the adaptive process and change frequently after the system converges.

Methods proposed in [51] and [58] utilize property 1, whereas techniques introduced in [53] and [57] utilize both of the properties.

All these algorithms introduced above perform well under certain conditions. However, from the perspective of robustness to high level noise interference, they all have disadvantages. It is clear to see from Table 5.1 that algorithms in [51], [52], [53] and [57] are sensitive to high level noise, since the instantaneous value $e(n)$ is used in their implementations, and can be contaminated by the noise, while the method in [54] needs the noise signal to be uncorrelated, and the method in [55] assumes that the noise is Gaussian or Gaussian-like. The algorithm proposed in [58] is more attractive as compared with other methods, but a low noise condition is necessary for the approximation used in the derivation. For high noise conditions, typically 0dB SNR or lower, the parameter choice guideline provided in [58] will fail since the approximation used in the derivation is no longer reasonable, and it will be very difficult to find proper parameters. To enhance the convergence rate of the LMS algorithm in high noise conditions, new VSSLMS algorithms

are needed.

Next the theoretically optimal VSSLMS algorithm provided in [56] will be introduced, which gives deeper insight into the selection of the step size at each iteration.

5.2 A theoretically optimal VSSLMS algorithm

In [56] a theoretically optimal VSSLMS algorithm, namely Mader's algorithm is given. To make the formulation consistent, Mader's algorithm will be derived based on the LMS algorithm in this section rather than the normalized LMS (NLMS) algorithm as in its original derivation in [56].

To derive Mader's method, the following assumptions are utilized:

A.5.2.1. The step size $\mu(n)$ is independent of $e(n)$, $\mathbf{x}(n)$ and $\mathbf{v}(n)$.

A.5.2.2. The noise signal $t(n)$ is independent of the excess error signal $\xi(n)$.

A.5.2.3. The input signal is statistically stationary, and the term $\|\mathbf{x}(n)\|^2$ can be approximated by a constant $L\sigma_x^2$.

The optimal step size of the LMS algorithm formulated in [56] is defined as the value which can obtain the largest decrease of the MSD $E\{\|\mathbf{v}(n)\|^2\}$ at each iteration. The largest decrease of the MSD is obtained by maximizing $E\{\|\mathbf{v}(n)\|^2\} - E\{\|\mathbf{v}(n+1)\|^2\}$. Substituting equation (5.1.3) into (5.1.5) yields

$$\mathbf{v}(n+1) = \mathbf{v}(n) - \mu(n)e(n)\mathbf{x}(n) \tag{5.2.1}$$

By using equation (5.2.1) and assumption *A.5.2.1*, the term $E\{\|\mathbf{v}(n)\|^2\}-$

$E\{\|\mathbf{v}(n+1)\|^2\}$ can be expanded as

$$E\{\|\mathbf{v}(n)\|^2\} - E\{\|\mathbf{v}(n+1)\|^2\} = -\mu^2(n)E\{e^2(n)\|\mathbf{x}(n)\|^2\} + 2\mu(n)E\{e(n)\mathbf{v}^T(n)\mathbf{x}(n)\} \quad (5.2.2)$$

Substituting equation (5.1.6) into (5.2.2) yields

$$E\{\|\mathbf{v}(n)\|^2\} - E\{\|\mathbf{v}(n+1)\|^2\} = -\mu^2(n)E\{e^2(n)\|\mathbf{x}(n)\|^2\} + 2\mu(n)E\{e(n)\xi(n)\} \quad (5.2.3)$$

From (5.2.3) it is straightforward to obtain that the largest decrease of the MSD is achieved when the step size is set to the value

$$\mu_{opt}(n) = \frac{E\{e(n)\xi(n)\}}{E\{e^2(n)\|\mathbf{x}(n)\|^2\}} \quad (5.2.4)$$

Utilizing assumptions *A.5.2.2* and *A.5.2.3*, the following optimal step size value is obtained by using (5.1.4) in (5.2.4) [56]

$$\mu_{opt}(n) \approx \frac{E\{\xi^2(n)\}}{\|\mathbf{x}(n)\|^2 E\{e^2(n)\}} = \frac{E\{e^2(n)\} - E\{t^2(n)\}}{\|\mathbf{x}(n)\|^2 E\{e^2(n)\}} \quad (5.2.5)$$

Unfortunately, neither the noise signal $t(n)$ nor its power are accessible, which makes this algorithm impractical. In [56], many methods have been discussed to estimate the optimal step size by some further signal detection techniques, such as the detection of the power of $t(n)$, which is the remote excitation signal within the context of adaptive noise cancellation (ANC). Since signal detection techniques are not the keystone in this chapter, the readers who are interested in this topic can refer to a key work in the literature [56].

The theoretically optimal step size value formulated in (5.2.5) gives a general guideline for the design of the VSSLMS algorithms. From this formulation, Shin's method [58] can be obtained directly by using some approximations. Utilizing (5.1.4) and assumption *A.5.2.2*, the theoretically optimal step size in (5.2.5) can be rewritten as

$$\mu_{opt}(n) \approx \frac{E\{\xi^2(n)\}}{\|\mathbf{x}(n)\|^2(E\{t^2(n)\} + E\{\xi^2(n)\})} \tag{5.2.6}$$

Using assumption *A.5.2.3* equation (5.2.6) can be approximated as

$$\mu_{opt}(n) \approx \frac{E\{\frac{\xi(n)\mathbf{x}^T(n)}{\|\mathbf{x}(n)\|^2}\frac{\xi(n)\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}\}}{(\frac{E\{t^2(n)\}}{\|\mathbf{x}(n)\|^2} + E\{\frac{\xi(n)\mathbf{x}^T(n)}{\|\mathbf{x}(n)\|^2}\frac{\xi(n)\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}\})\|\mathbf{x}(n)\|^2} \tag{5.2.7}$$

Shin's algorithm can then be obtained by approximating the term $\frac{E\{t^2(n)\}}{\|\mathbf{x}(n)\|^2}$ as c, and the term $E\{\frac{\xi(n)\mathbf{x}^T(n)}{\|\mathbf{x}(n)\|^2}\frac{\xi(n)\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}\}$ as $\|\overline{\mathbf{g}}_n(n)\|^2$, where c and $\overline{\mathbf{g}}_n(n)$ are shown in Table 5.1[1]. As shown by the authors, this approximation is reasonable for low noise conditions (in the simulation in [58], the SNR is 30dB).

Although Shin's algorithm is more attractive in practice since it can provide an approximately optimal step size value for each iteration, and performs better than other VSSLMS algorithms [52] and [54], the low noise condition is necessary for the approximation $E\{\frac{\xi(n)\mathbf{x}^T(n)}{\|\mathbf{x}(n)\|^2}\frac{\xi(n)\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}\} \approx \|\overline{\mathbf{g}}_n(n)\|^2$. For high noise conditions, this approximation is seriously biased, thus the parameter choice guideline provided in [58], that $\mu_{max} = 1$ and $c = \frac{\sigma_i^2}{L\sigma_x^2}$ will no longer result in good performance, and choosing proper parameters will be a serious problem for this algorithm.

It has been shown in both [57] and [58] that the smoothing operation performed on the gradient vector, such as the operation per-

[1]Note that μ_{max} in Shin's algorithm is suggested to be unity.

formed on $\overline{\mathbf{g}}_n(n)$ in Shin's algorithm can reduce the interference of the noise. Whether the noise level is low or high, the squared norm of the smoothed gradient vector will be large initially, and small at steady-state, thus it is a good measure that can be used to control the step size. Motivated by this property of the gradient vector, two new VSSLMS algorithms are provided in the following sections, which are designed for applications with high level noise interference.

5.3 A new VSSLMS algorithm with robustness to statistically stationary noise

A new VSSLMS algorithm with robustness to statistically stationary noise is described in this section. In this algorithm, the gradient vector is smoothed by using a first order filter to reduce the disturbance of the noise signal. Then the step size is controlled to be proportional to the squared norm of the smoothed gradient vector. The theoretical steady-state performance analysis and guideline for the parameter choice are also provided. The proposed algorithm will be compared with Mathews' algorithm [53], Ang's algorithm [57] and Shin's method [58] by simulations, to show its advantages.

5.3.1 Algorithm formulation

The update of the step size of the proposed VSSLMS algorithm can be formulated as follows:

$$\overline{\mathbf{g}}(n) = \beta\overline{\mathbf{g}}(n-1) + (1-\beta)\mathbf{x}(n)e(n) \tag{5.3.1}$$

$$\mu(n) = P\|\overline{\mathbf{g}}(n)\|^2 \tag{5.3.2}$$

where $\overline{\mathbf{g}}(n)$ is the smoothed gradient vector, β is the smoothing parameter which is set to be very close to unity to apply sufficient time smoothing, P is a positive constant and can be chosen easily according to the analysis in the next subsection.

The motivation of the proposed algorithm is as follows: to develop a VSSLMS algorithm, the most important thing is to measure the proximity of the adaptive process to the desired solution. An ideal measure of the adaptive process is the MSD $E\{\|\mathbf{v}(n)\|^2\}$. According to the formulation in [58], with a statistically stationary input signal, the squared norm of the smoothed gradient vector, which is formulated in (5.3.1), can track the variation of the MSD. Thus, it is a good measure of the proximity of the adaptive process, and suitable to control the step size. As will be shown by simulations, the proposed algorithm performs well in both low and high noise conditions.

5.3.2 Steady-state performance analysis

An approximate steady-state performance analysis for the proposed VSSLMS algorithm is provided in this subsection. For convenience of analysis, several assumptions are utilized:

A.5.3.1. The input signal is zero-mean white statistically stationary. The noise signal is zero-mean statistically stationary and independent of the input signal.

A.5.3.2. At steady-state the excess error is much smaller as compared with the noise signal, and therefore the error signal $e(n)$ is approximately equal to the noise signal $t(n)$.

Assumption *A.5.3.1* is a general assumption for the analysis of the VSSLMS algorithms. Assumption *A.5.3.2* is only true when the step

size is very small. Using these assumptions gives insight into the algorithm and provides a guideline for the parameter choice of the algorithm.

Since the squared norm of the smoothed normalized gradient vector $\|\overline{\mathbf{g}}(n)\|^2$ is the key term for the proposed algorithm, a steady-state performance analysis for this term is performed first. From (5.3.1) the following equation is obtained

$$\overline{\mathbf{g}}(n) = (1-\beta)\sum_{i=1}^{n}\beta^{n-i}\mathbf{g}(i) \tag{5.3.3}$$

assuming $\overline{\mathbf{g}}(0) = \mathbf{0}$ and denoting $\mathbf{g}(i) = e(i)\mathbf{x}(i)$. The expected value of the squared norm of the smoothed gradient vector can then be obtained

$$E\{\|\overline{\mathbf{g}}(n)\|^2\} = (1-\beta)^2\sum_{i=1}^{n}\sum_{j=1}^{n}C(i,j) \tag{5.3.4}$$

where $C(i,j)$ is defined as

$$C(i,j) = E\{\beta^{n-i}\mathbf{g}^T(i)\beta^{n-j}\mathbf{g}(j)\} \tag{5.3.5}$$

When n approaches infinity, the term β^{n-i} in (5.3.5) approaches zero if i or j is finite. So in the calculation of $E\{\|\overline{\mathbf{g}}(\infty)\|^2\}$, the term $C(i,j)$ can be ignored when i or j is finite. The following analysis will only consider this term at steady-state, i.e., i and j are both steady-state time indexes.

The term $C(i,j)$ when $i = j$ is considered first. From assumption *A.5.3.2* the error signal $e(n)$ can be approximated as the noise signal

$$e(i) \approx t(i) \tag{5.3.6}$$

With (5.3.6) the gradient vector $\mathbf{g}(i)$ can also be approximately written as

$$\mathbf{g}(i) \approx t(i)\mathbf{x}(i) \tag{5.3.7}$$

Substituting this formulation into (5.3.5) yields

$$C(i,i) \approx E\{\beta^{2n-2i}\mathbf{x}^T(i)\mathbf{x}(i)t^2(i)\} \tag{5.3.8}$$

With assumption *A.5.3.1*, equation (5.3.8) becomes

$$C(i,i) \approx \beta^{2n-2i}L\sigma_x^2\sigma_t^2 \tag{5.3.9}$$

When $i \neq j$ similar derivation can be performed, and the following equation can be obtained

$$C(i,j) \approx 0 \text{ when } i \neq j \tag{5.3.10}$$

Substituting (5.3.9) and (5.3.10) into (5.3.4) yields

$$\lim_{n\to\infty} E\{\|\overline{\mathbf{g}}(n)\|^2\} \approx (1-\beta)^2 \sum_{i=s}^{n} \beta^{2(n-i)}L\sigma_x^2\sigma_t^2 \tag{5.3.11}$$

where s is the time index at which when $i \geq s$ the system is assumed at steady-state. Equation (5.3.11) can be simplified as:

$$E\{\|\overline{\mathbf{g}}(\infty)\|^2\} \approx \frac{(1-\beta)}{(1+\beta)}L\sigma_x^2\sigma_t^2 \tag{5.3.12}$$

Based on the steady-state formulation $E\{\|\overline{\mathbf{g}}(\infty)\|^2\}$ in (5.3.12) the steady-state performance of the proposed algorithm can be easily obtained. Taking the statistical expectation from both sides of (5.3.2) the

following equation is obtained

$$E\{\mu(n)\} = PE\{\|\bar{\mathbf{g}}(n)\|^2\} \tag{5.3.13}$$

Substituting (5.3.12) into (5.3.13) yields

$$E\{\mu(\infty)\} \approx \frac{P(1-\beta)L\sigma_x^2\sigma_t^2}{(1+\beta)} \tag{5.3.14}$$

The steady-state EMSE of the LMS algorithm with a fixed step size value μ_{LMS} can be formulated as [15]

$$J_{ex,LMS}(\infty) = \frac{\mu_{LMS}L\sigma_x^2\sigma_t^2}{2-\mu_{LMS}L\sigma_x^2} \tag{5.3.15}$$

Note that at steady-state, $\mu(\infty)$ can be approximately deemed as a fixed value, and the adaptation of the proposed algorithm will be similar to that of the fixed step size LMS algorithm. Assuming that at steady-state the step size of the proposed algorithm is very small, and $\mu(\infty)L\sigma_x^2 \ll 2$, with equation (5.3.15) the EMSE of the proposed algorithm can be approximated as

$$J_{ex,VSS}(\infty) \approx \frac{1}{2}E\{\mu(\infty)\}L\sigma_x^2\sigma_t^2 \tag{5.3.16}$$

Substituting (5.3.14) into (5.3.16) the steady-state EMSE for the proposed VSSLMS algorithm is then obtained:

$$J_{ex,VSS}(\infty) \approx \frac{P(1-\beta)L^2\sigma_x^4\sigma_t^4}{2(1+\beta)} \tag{5.3.17}$$

The choice of the parameter P is crucial for the application of the proposed algorithm. To choose this parameter, a desired EMSE

$J_{ex,VSS}(\infty)$ is determined first according to the application. With $J_{ex,VSS}(\infty)$ the parameter P can be determined according to (5.3.17):

$$P = \frac{2J_{ex,VSS}(\infty)(1+\beta)}{(1-\beta)L^2\sigma_x^4\sigma_t^4} \tag{5.3.18}$$

Two simulations will be performed in the next subsection to support the analysis and show the advantages of the proposed algorithm.

5.3.3 Simulation

Two simulations are performed in this subsection to demonstrate the advantages of the proposed algorithm. In both simulations the proposed algorithm will be compared with Mathews' method, Ang's method and Shin's method. The implementations of all these algorithms can be found in Table 5.1. The system identification model is assumed in both simulations. The performance comparison is performed on the basis of the following measures: 1. The time evolution of the step size. 2. The EMSE which is defined as $E\{\xi^2(n)\}$. The results of both simulations are obtained by 100 Monte Carlo trials.

In the first simulation, a five-point FIR filter $\mathbf{w}_{opt}$=[0.1 0.3 0.5 0.3 0.1] is considered to be identified [53]. The input signal $x(n)$ is a pseudo random, zero-mean, unit variance Gaussian process. The noise signal $t(n)$ is a pseudo random, zero-mean, Gaussian process uncorrelated with $x(n)$ and scaled to make the SNR 20dB. The initial step sizes and adaptive filter vectors of all algorithms are set to be zero. The parameter ρ for Mathews' method [53] is set to 5×10^{-4}. The smoothing parameter β for Ang's method [57], Shin's method [58] and the proposed method is set to 0.99. The parameter γ for Ang's method is

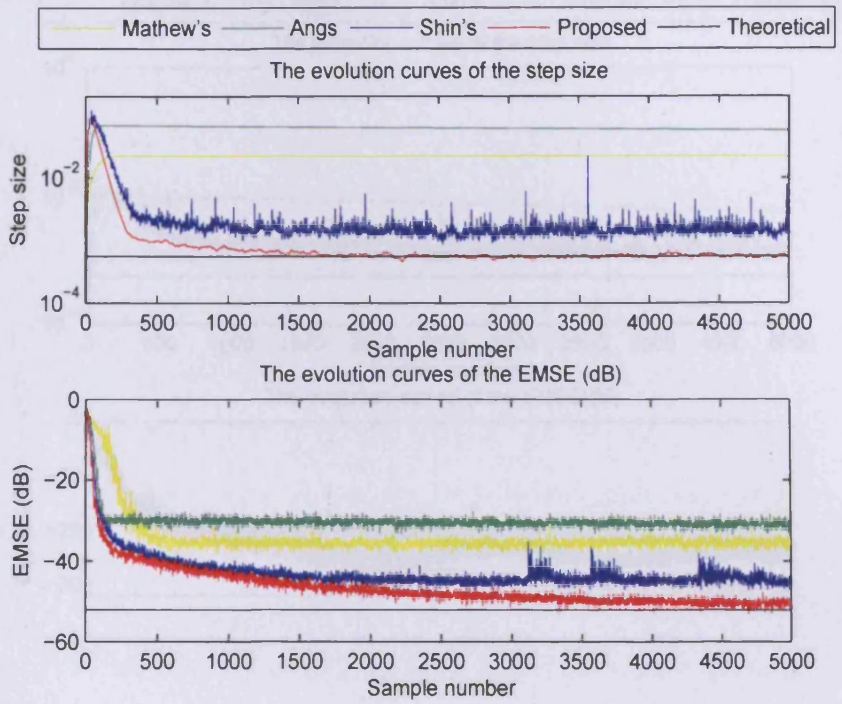

Figure 5.1. Monte Carlo averaged simulation results of the step size and EMSE for different algorithms when SNR=20dB

set to 2×10^{-4}. For Shin's method, the parameter c is set to $\frac{\sigma_i^2}{L\sigma_x^2}$ and the parameter μ_{max} is set to 0.5[2]. The parameter P for the proposed algorithm is empirically set to 5, and produces a steady-state EMSE as predicted by (5.3.17). The evolutions of the step sizes and the EMSE curves are shown in Fig. 5.1.

The theoretical values of the step size and the EMSE of the proposed algorithm according to (5.3.14) and (5.3.17) are also plotted. From Fig. 5.1 it is clear to see that both Shin's method and the proposed method perform better than the other two algorithms in this case. The step sizes of both Mathews' method and Ang's method converge slowly, which results in a slow convergence rate of the system. Although Shin's method

[2]According to the guideline in [58], μ_{max} should be set to 1. However, it has been observed from simulations that the value 0.5 is much better than 1, thus 0.5 is chosen in both simulations.

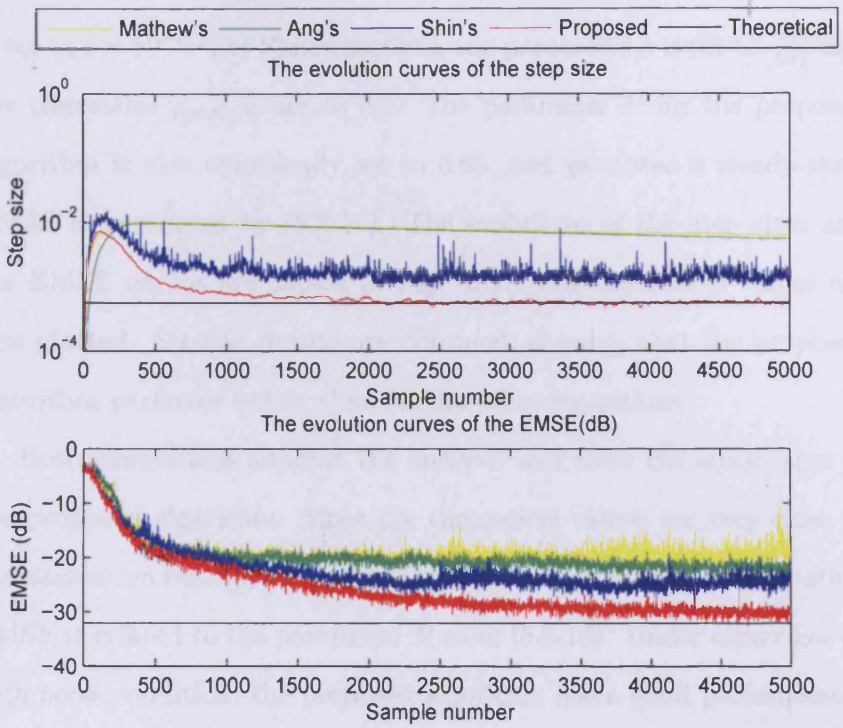

Figure 5.2. Monte Carlo averaged simulation results of the step size and EMSE for different algorithms when SNR=0dB

is derived from the theoretically optimal step size VSSLMS algorithm, the approximation used in the derivation influences its performance. As can be seen in Fig. 5.1, the proposed algorithm has a similar convergence rate as compared with Shin's method, but the steady-state EMSE is smaller. The theoretical value of the proposed algorithm is very close to the simulation results, which confirms the analysis in the previous subsection.

The set up of the second simulation is similar to the first simulation except the noise signal is scaled to make the SNR 0dB. In this simulation, the parameter ρ for Mathews' method is set to 1×10^{-4}. The smoothing parameter β for Ang's method, Shin's method and the proposed algorithm is set to 0.99. The parameter γ for Ang's method

is set to 2×10^{-6}. For Shin's method, the parameter c is set to $\frac{\sigma_i^2}{L\sigma_x^2}$ and the parameter μ_{max} is set to 0.5. The parameter P for the proposed algorithm is also empirically set to 0.05, and produces a steady-state EMSE as predicted by (5.3.17). The evolutions of the step sizes and the EMSE curves are shown in Fig. 5.2. The theoretical values are also plotted. Similar results are obtained, showing that the proposed algorithm performs better than all the other algorithms.

Both simulations support the analysis and show the advantages of the proposed algorithm. Since the theoretical values are very close to the simulation results, the conclusion can be made that the steady-state EMSE is related to the parameter P as in (5.3.18). Under either low or high noise condition, the proposed algorithm has a good performance.

Note that according to the simulations not included in this thesis, although the performance of Shin's algorithm can be improved slightly by different parameter settings of c and μ_{max}, it is very difficult to find out the optimal parameter set. From the perspective of both good performance and easy implementation, the proposed algorithm is preferred.

All the algorithms discussed above are based on the assumption that the noise signal is statistically stationary. A new VSSLMS algorithm with robustness to statistically nonstationary noise will be introduced in the next section.

5.4 A new VSSLMS algorithm with robustness to statistically nonstationary noise

In this section a new VSSLMS algorithm is proposed which is designed to be robust to high level, e.g., SNR= 0dB or worse statistically nonstationary noise interference. The step size of the proposed VSSLMS algorithm is controlled by the normalized square Euclidean norm of the averaged gradient vector, and is henceforth referred to as the NSVSSLMS algorithm.

5.4.1 Algorithm formulation

The NSVSSLMS algorithm is motivated by the sum method which is proposed in [48]. This sum method is designed to be suitable for statistically nonstationary input and noise signals and can be formulated as follows:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu_{\text{sum}} e(n)\mathbf{x}(n)}{L[\hat{\sigma}_e^2(n) + \hat{\sigma}_x^2(n)]} \tag{5.4.1}$$

where μ_{sum} is the step size for this sum method, $\hat{\sigma}_e^2(n)$ and $\hat{\sigma}_x^2(n)$ are time-varying estimations of the output error signal variance and the input signal variance respectively [48]. As explained in [48], the step size in (5.4.1) is adjusted by the input and output error variance automatically, which reduces the influence brought by the fluctuation of the input and the noise signals. However, it is based on a constant convergence rate. Similar to the case of the LMS algorithm, a variable step-size algorithm is also necessary to obtain both a fast convergence rate and a small steady-state MSE.

According to the properties of the sum method and the VSSLMS algorithm proposed in the previous section, a new NSVSSLMS algorithm

is designed for applications in which the noise level is high and statistically nonstationary. The update of the step size of this NSVSSLMS algorithm can be formulated as follows:

$$\mu_{\text{NSVSS}}(n) = \frac{P \|\bar{\mathbf{g}}(n)\|_2^2}{\{L[\hat{\sigma}_e^2(n) + \hat{\sigma}_x^2(n)]\}^2} \tag{5.4.2}$$

where $\mu_{\text{NSVSS}}(n)$ is the time-varying step size, $\bar{\mathbf{g}}(n)$ is the smoothed gradient vector, as can be seen in (5.3.1), and P is a positive constant which can be easily chosen according to the analysis in the next subsection.

This algorithm is motivated as follows: as shown in the previous section the squared norm of the smoothed gradient vector which is formulated in (5.3.1) is suitable to control the step size. The term $L[\hat{\sigma}_e^2(n) + \hat{\sigma}_x^2(n)]$ in (5.4.2) is motivated by the sum method formulated in (5.4.1). The square of this term, as a novel normalization for the step size, is designed to make the steady-state EMSE of the proposed algorithm robust to the noise signal, according to the analysis in the next subsection. This algorithm can be deemed as a variable step-size version of the sum method.

It will be shown in the simulations that as compared with the sum method, the proposed algorithm has both a fast convergence rate and robustness to high level statistically nonstationary noise signal. Furthermore, the parameter P in the proposed algorithm can be easily determined according to the following steady-state performance analysis.

5.4.2 Steady-state performance analysis

For convenience of analysis, the same assumptions as in the previous section are made. The assumption that the noise is statistically stationary is justified as many signals such as speech can be assumed statistically stationary over a certain interval.

In the analysis in the previous section, the steady-state value of the squared norm of the smoothed gradient vector $\|\bar{\mathbf{g}}(n)\|_2^2$ is formulated in (5.3.12). Furthermore, since the term $\{L[\hat{\sigma}_x^2(n)+\hat{\sigma}_t^2(n)]\}^2$ changes very slowly with statistically stationary input and noise signals, it is assumed to be a constant during the iteration. Taking statistical expectation on both sides of (5.4.2) and utilizing (5.3.12) the steady-state value of the step size is obtained

$$E\{\mu_{\text{NSVSS}}(\infty)\} \approx \frac{P(1-\beta)L\sigma_x^2\sigma_t^2}{(1+\beta)\{L[\hat{\sigma}_x^2(n)+\hat{\sigma}_t^2(n)]\}^2}. \tag{5.4.3}$$

Similar to the analysis in the previous section, the step size of the NSVSSLMS algorithm is assumed to be very small at steady-state, and $\mu_{\text{NSVSS}}(\infty)L\sigma_x^2 \ll 2$, the EMSE of the proposed algorithm can then be formulated as

$$J_{\text{ex,NSVSS}}(\infty) \approx \frac{1}{2}E\{\mu_{\text{NSVSS}}(\infty)\}L\sigma_x^2\sigma_t^2. \tag{5.4.4}$$

Substituting (5.4.3) into (5.4.4) the steady-state EMSE for the proposed NSVSSLMS algorithm can be obtained:

$$J_{\text{ex,NSVSS}}(\infty) \approx \frac{P(1-\beta)L^2\sigma_x^4\sigma_t^4}{2(1+\beta)\{L[\hat{\sigma}_x^2(n)+\hat{\sigma}_t^2(n)]\}^2}. \tag{5.4.5}$$

Since $\hat{\sigma}_t^2(n) \approx \sigma_t^2$, the following equation is obtained from (5.4.5)

$$\lim_{\sigma_t^2 \to \infty} J_{\text{ex,NSVSS}}(\infty) \approx \frac{P(1-\beta)\sigma_x^4}{2(1+\beta)}. \tag{5.4.6}$$

It can be clearly seen from (5.4.6) that the EMSE obtained by the proposed algorithm will be independent of the noise signal $t(n)$ when the variance of the noise signal is very large. Although the analysis is based on the assumption that the noise signal is statistically stationary, an approximate indication of its general performance is also obtained. For some statistically nonstationary noise signals, such as speech, they can be deemed as approximately short-term statistically stationary over some short intervals. When the variance of some intervals of the noise signal is much higher than the input signal at steady-state, the EMSE will be independent of the variance of the noise signal; thus, the proposed algorithm is robust for applications with high level statistically nonstationary noise signals.

The choice of the parameter P is very important for the application of the NSVSSLMS algorithm. Note that (5.4.6) also gives an upper bound of the steady-state EMSE of the proposed algorithm with the variation of the noise variance. To choose this parameter, an upper bound value of $J_{\text{ex,NSVSS}}$ is determined according to the application. With this value and the variance of the input signal, P can be determined directly according to (5.4.6):

$$P = \frac{J_{\text{ex,NSVSS, max}} 2(1+\beta)}{(1-\beta)\sigma_x^4} \tag{5.4.7}$$

where $J_{\text{ex,NSVSS, max}}$ is the upper bound value of the EMSE.

If the maximum short-interval variance of the statistically nonsta-

tionary noise signal can be obtained, a more accurate criterion for the choice of P similar to (5.4.7) can be obtained according to (5.4.5). In practice, since the noise variance is not infinite, the parameter P can be a little larger than the value obtained from (5.4.7).

In the next subsection, all the above analysis and discussion will be supported by simulations in the context of a statistically nonstationary noise signal.

5.4.3 Simulation

In this subsection the performance of the sum method and the proposed algorithm is compared. The input signal $x(n)$ is a pseudo-random, zero-mean unit-variance Gaussian signal with a length of 100,000 samples. The noise signal $t(n)$ is the first $100,000$ samples of a speech signal which is available from http://www.voiptroubleshooter.com/open_speech/american.html, and the file name is "OSR_us_000_0016_8k.wav". This noise signal is scaled to make the average SNR 0dB over the entire observation. The noise signal and one representation of the input signal can be seen in Fig. 5.3.

The primary signal $d(n)$ is obtained as follows:

$$d(n) = x(n) * h(n) + t(n) \tag{5.4.8}$$

where $h(n)$ is the causal optimal filter obtained by

$$h(n) = \begin{cases} e^{-0.05(n-1)}r(n), & n = 1, ..., 100 \\ 0 & otherwise \end{cases} \tag{5.4.9}$$

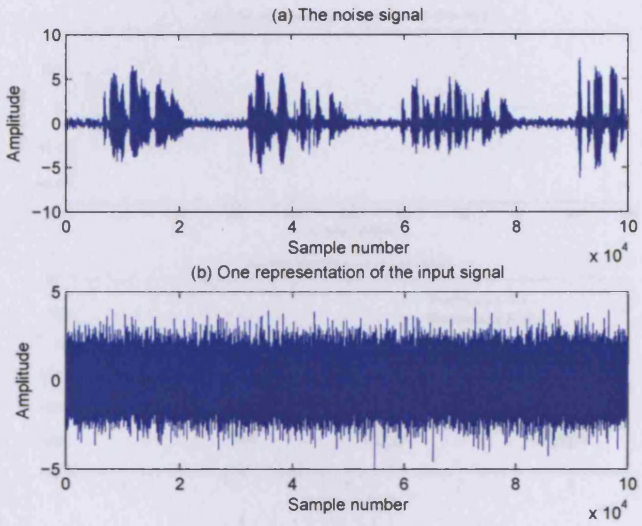

Figure 5.3. The noise signal (a) and one representation of the input signal (b).

where $r(n)$ is drawn from a zero mean unit variance Gaussian sequence. One representation of the nonzero terms of $h(n)$ can be seen in Fig. 5.4(a).

In this simulation the proposed algorithm will be compared with the sum method with different step sizes 0.1 and 0.02. The initial step sizes and adaptive filter vectors of the proposed algorithm are set to zero. The parameter β for the proposed algorithm is set to 0.999 to perform a sufficient smoothing operation. The parameter P in the proposed algorithm is empirically set to 80, and produces a steady-state EMSE as predicted by (5.4.6). The parameter sets for the proposed algorithm are chosen to make its initial convergence rate approximately equal to that of the sum method with a step size 0.1. The estimates $\hat{\sigma}_e^2(n)$ and $\hat{\sigma}_x^2(n)$ used in the sum algorithm and the proposed algorithm are obtained by smoothing the input and error signals as

$$\hat{\sigma}_e^2(n) = 0.99\hat{\sigma}_e^2(n-1) + (1-0.99)e^2(n) \qquad (5.4.10)$$

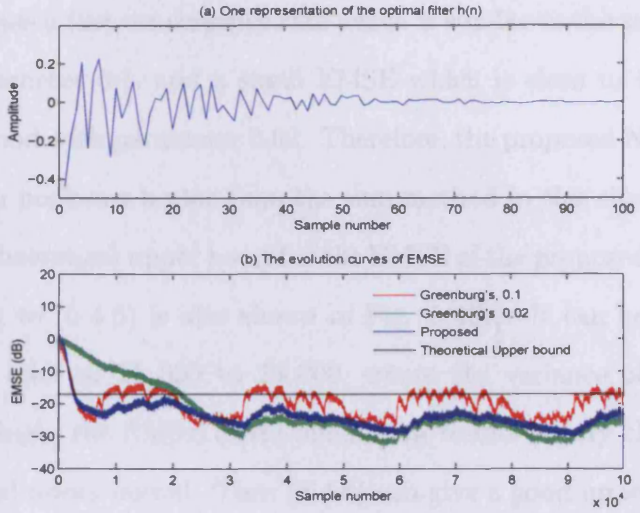

Figure 5.4. One representation of the optimal filter (a) and the Monte Carlo averaged evolution curves of the EMSE for the sum method and the proposed NSVSSLMS algorithms (b).

and

$$\hat{\sigma}_x^2(n) = 0.99\hat{\sigma}_x^2(n-1) + (1-0.99)x^2(n) \tag{5.4.11}$$

The initial values of $\hat{\sigma}_e^2(n)$ and $\hat{\sigma}_x^2(n)$ are set to zero and unity respectively. The evolutions of the EMSE curves for all the experiments are shown in Fig. 5.4(b). The results are obtained over 200 Monte Carlo trials of the same experiment.

It is clear to see in Fig. 5.4(b) that the proposed algorithm has an EMSE convergence rate similar to that of the sum method with a parameter 0.1 at the early state of the process. The EMSE of both methods converges to -20dB at approximately 3,000 samples. However, the EMSE of the sum method with parameter 0.1 fluctuates greatly with the variation of the noise signal energy. The performance of the sum method with parameter 0.02 has a small EMSE and slight fluctuation of the EMSE, but the convergence rate is very slow. The proposed al-

gorithm has a fast convergence rate which is similar to the sum method with parameter 0.1, and a small EMSE which is close to that of the sum method with parameter 0.02. Therefore, the proposed NSVSSLMS algorithm performs better than the sum method in this simulation.

The theoretical upper bound of the EMSE of the proposed algorithm according to (5.4.6) is also shown in Fig. 5.4(b). It can be seen that over the interval 15,000 to 20,000, where the variance of the noise signal is high, the EMSE of the simulation results is very close to this theoretical upper bound. Thus (5.4.6) can give a good upper bound of the steady-state EMSE for the proposed algorithm, and the conclusion can be made that with a given upper bound of the steady-state EMSE, the parameter P can be properly chosen according to (5.4.7).

Note that all the analysis and simulations are based on a white input signal. When the input signal is correlated, the analysis results obtained from (5.4.3) and (5.4.5) are both incorrect, and smaller than the practical results. In this case, the parameter P should be chosen smaller than the value obtained from (5.4.6). Finally, if both input and noise signals are statistically nonstationary signals, the smoothed gradient vector can not measure the proximity of the adaptive process, and the proposed algorithm has no advantage as compared with the sum method.

Although the proposed algorithm is only compared with the sum method in this section, many experiments which are not included in this section have been performed, and it has been shown that all the other existing VSSLMS algorithms perform poorly with such a nonstationary noise signal, since their EMSE is proportional to the noise variance.

5.5 Conclusion

In this chapter, an overview of typical existing VSSLMS algorithms and an introduction for a theoretically optimal VSSLMS algorithm has been given. Two new VSSLMS algorithms have also been proposed, which are designed for high level noise conditions. Simulations show that these algorithms can obtain both a fast convergence rate and a small EMSE with robustness to statistically stationary or nonstationary noise signals, and perform better as compared with other existing VSSLMS algorithms. Both methods may be potentially used in many applications of adaptive filtering.

Chapter 6

VARIABLE TAP-LENGTH LMS ALGORITHMS

The least mean square (LMS) algorithm has been widely used in many applications. A detailed introduction of the LMS algorithm is given in Chapter 3. Variable step-size LMS (VSSLMS) algorithms are discussed in Chapter 5. In this chapter, the topic of variable tap-length LMS (VTLMS) algorithms will be investigated. This topic is motivated by the assumption of a fixed tap-length, as in many applications of the LMS algorithm, potentially leading to degraded performance. In certain situations, for example, the tap-length of the optimal filter can be unknown or possibly variable. According to the analysis in [1] and [59], the mean square error (MSE) of the adaptive filter is likely to increase if the tap-length is undermodelled. To avoid such a situation, a sufficiently large filter tap-length is needed. However, the computational cost and the misadjustment noise are proportional to the tap-length, thus variable tap-length LMS algorithms are needed to find a proper choice of the tap-length.

Several variable tap-length LMS algorithms have been proposed in recent years. Among existing variable tap-length LMS algorithms, some are designed to not only establish a suitable steady-state tap-length,

but also to speed up the convergence rate [1]. Other methods are more general and are designed to search for the optimal filter tap-length at steady-state [60] [61] [62] [63]; a summary of these works is given in [12]. As analyzed in [12], the fractional tap-length (FT) algorithm is more robust and has lower computational complexity when compared with other methods. Since the methods in both [60] and [61] can be deemed as special cases of the method proposed in [12], the work on VTLMS algorithms in this chapter will mainly focus on the analysis and improvement of the FT algorithm. A new VTLMS algorithm to model an exponential decay impulse response is also provided in this chapter.

The organization of this chapter is as follows: an introduction of the FT algorithm is given in Section 6.1. The steady-state performance analysis of the FT algorithm and the guideline for the parameters choice are provided in Section 6.2. To improve the performance of the FT algorithm in high noise conditions, a convex combination structure is utilized in Section 6.3. In Section 6.4, a practical optimal variable tap-length LMS algorithm is proposed, which is designed for the applications where the optimal filter has an exponential decay impulse response. Section 6.5 concludes this chapter.

6.1 The FT VTLMS algorithm

The FT VTLMS algorithm is designed to find the optimal filter tap-length. In agreement with most approaches used to derive algorithms for adaptive filtering, the design problem is related to the optimization of a certain criterion that is dependent on the tap-length. For convenience, the LMS algorithm is formulated within a system identi-

fication framework, in which the unknown filter $\mathbf{c}_{L_{opt}}$ has an unknown tap-length L_{opt} which is to be identified. In this model, the desired signal $d(n)$ is represented as

$$d(n) = \mathbf{x}_{L_{opt}}^T(n)\mathbf{c}_{L_{opt}} + v(n) \tag{6.1.1}$$

where $\mathbf{x}_{L_{opt}}(n)$ is the input vector with a tap length of L_{opt}, $v(n)$ is a zero mean additive noise term uncorrelated with the input, n denotes the discrete time index, and $(\cdot)^T$ denotes the transpose operation. In the work in this chapter all quantities are assumed to be real valued.

For convenience of description the tap-length of the adaptive filter is assumed to be a fixed value at steady-state and denoted by L; $\mathbf{w}_L$ and $\mathbf{x}_L(n)$ are respectively the corresponding steady-state adaptive filter vector and input vector. Also, the segmented steady-state error is defined as [12]

$$e_M^{(L)}(n) = d(n) - \mathbf{w}_{L,1:M}^T\mathbf{x}_{L,1:M}(n), \text{ as } n \to \infty \tag{6.1.2}$$

where $1 \leq M \leq L$, $\mathbf{w}_{L,1:M}$ and $\mathbf{x}_{L,1:M}(n)$ are respectively vectors consisting of the first M coefficients of the steady-state filter vector $\mathbf{w}_L$ and the input vector $\mathbf{x}_L(n)$. The mean square of this segmented steady-state error is defined as $\xi_M^{(L)} = E\{(e_M^{(L)}(n))^2\}$. The underlying basis of the FT method is to find the minimum value of L satisfying [12]:

$$\xi_{L-\Delta}^{(L)} - \xi_L^{(L)} \leq \varepsilon \tag{6.1.3}$$

where Δ is a positive integer less than L (the guideline for the selection of this parameter is given in Section 6.2.2), and ε is a small positive

value determined by the system requirements. The minimum L that satisfies (6.1.3) is then chosen as the optimum tap-length. A detailed description of this criterion and another similar criterion can be found in [12].

Gradient-based methods can be used to estimate L on the basis of (6.1.3). However, the tap-length that should be used in the adaptive filter structure must be an integer, and this constrains the adaptation of the tap length. Different approaches have been applied to solve this problem [60] [61] [62] [63] [12]. In [12], the concept of "pseudo fractional tap-length", denoted by $l_f(n)$, is utilized to make instantaneous tap-length adaptation possible. The update of the fractional tap-length is as follows:

$$l_f(n+1) = (l_f(n) - \alpha) - \gamma \left[(e_{L(n)}^{(L(n))})^2 - (e_{L(n)-\Delta}^{(L(n))})^2 \right] \qquad (6.1.4)$$

where γ is the step size for the tap-length adaptation, and α is a positive leakage parameter [12]. As explained in [12], $l_f(n)$ is no longer constrained to be an integer, and the tap-length $L(n+1)$, which will be used in the adaptation of the filter weights in the next iteration, is obtained from the fractional tap-length $l_f(n)$ as follows:

$$L(n+1) = \begin{cases} \lfloor l_f(n) \rfloor & \text{if } |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases} \qquad (6.1.5)$$

where $\lfloor . \rfloor$ is the floor operator, which rounds down the embraced value to the nearest integer and δ is a small integer.

Next a steady-state performance analysis based on the above formulation will be given.

6.2 Steady-state performance of the FT algorithm

In the FT algorithm, the filter coefficients are updated as

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e_{L(n)}^{(L(n))}(n)\mathbf{x}_{L(n)}(n) \tag{6.2.1}$$

where $\mathbf{w}_{L(n)}$ and $\mathbf{x}_{L(n)}$ are respectively the $L(n)$-tap adaptive filter vector and the input vector, μ is the positive step size for the update of the coefficients.

For convenience of analysis, a vector $\mathbf{c}_N$ is used to denote the unknown filter, where N is an integer larger than both the optimal tap-length L_{opt} and the maximum value of the variable tap-length sequence $L(n)$, and $\mathbf{c}_N$ is obtained by padding $\mathbf{c}_{L_{opt}}$ with zeros. This unknown filter vector $\mathbf{c}_N$ can be split into three parts:

$$\begin{pmatrix} \mathbf{c}' \\ \mathbf{c}'' \\ \mathbf{c}''' \end{pmatrix}$$

where $\mathbf{c}'$ is the part modelled by $\mathbf{w}'(n)$, $\mathbf{w}'(n) = \mathbf{w}_{L(n),1:L(n)-\Delta}(n)$, $\mathbf{c}''$ is the part modelled by $\mathbf{w}''(n)$, $\mathbf{w}''(n) = \mathbf{w}_{L(n),L(n)-\Delta+1:L(n)}(n)$ and $\mathbf{c}'''$ is the under-modelled part. The total coefficient error vector is denoted as $\mathbf{g}_N(n)$

$$\mathbf{g}_N(n) = \mathbf{c}_N - \mathbf{w}_N(n) \tag{6.2.2}$$

where $\mathbf{w}_N(n)$ is obtained by padding $\mathbf{w}_{L(n)}(n)$ with zeros. Therefore,

$\mathbf{g}_N(n)$ can be similarly split as

$$\begin{pmatrix} \mathbf{g}'(n) \\ \mathbf{g}''(n) \\ \mathbf{g}'''(n) \end{pmatrix}$$

The mean square deviation (MSD) between the optimal filter vector and the adaptive filter vector is given by $E\{\|\mathbf{g}_N(n)\|_2^2\}$, where $\|\cdot\|_2^2$ denotes the squared Euclidean distance.

For convenience of description, the input vector $\mathbf{x}_N(n)$ is split similarly to that of $\mathbf{c}_N(n)$ and $\mathbf{g}_N(n)$. With the above notation and substituting (6.1.1) and (6.2.2) into (6.1.2), and padding all the vectors in (6.1.1) and (6.1.2) with zeros to make their lengths equal to N yields

$$e_{L(n)}^{(L(n))}(n) = \begin{pmatrix} \mathbf{x}'(n) \\ \mathbf{x}''(n) \\ \mathbf{x}'''(n) \end{pmatrix}^T \begin{pmatrix} \mathbf{g}'(n) \\ \mathbf{g}''(n) \\ \mathbf{c}''' \end{pmatrix} + v(n) \tag{6.2.3}$$

and

$$e_{L(n)-\Delta}^{(L(n))}(n) = \begin{pmatrix} \mathbf{x}'(n) \\ \mathbf{x}''(n) \\ \mathbf{x}'''(n) \end{pmatrix}^T \begin{pmatrix} \mathbf{g}'(n) \\ \mathbf{c}'' \\ \mathbf{c}''' \end{pmatrix} + v(n) \tag{6.2.4}$$

The term $(e_{L(n)}^{(L(n))})^2 - (e_{L(n)-\Delta}^{(L(n))})^2$, which is the key term in the fractional tap-length update equation (6.1.4), can then be written as

$$\begin{aligned}(e_{L(n)}^{(L(n))})^2 - (e_{L(n)-\Delta}^{(L(n))})^2 = [&2v(n) + 2\mathbf{x}'^T(n)\mathbf{g}'(n) + \mathbf{x}''^T(n)\mathbf{g}''(n) \\ &+\mathbf{x}''^T(n)\mathbf{c}'' + 2\mathbf{x}'''^T(n)\mathbf{c}'''][\mathbf{x}''^T(n)\mathbf{g}''(n) - \mathbf{x}''^T(n)\mathbf{c}'']\end{aligned} \tag{6.2.5}$$

This term can be expanded as

$$
\begin{aligned}
(e_{L(n)}^{(L(n))})^2 - (e_{L(n)-\Delta}^{(L(n))})^2 = & \underbrace{2v(n)\mathbf{x}'^T(n)\mathbf{g}''(n)}_{A} - \underbrace{2v(n)\mathbf{x}'^T(n)\mathbf{c}''}_{B} \\
& + \underbrace{2\mathbf{x}'^T(n)\mathbf{g}'(n)\mathbf{x}'^T(n)\mathbf{g}''(n)}_{C} - \underbrace{2\mathbf{x}'^T(n)\mathbf{g}'(n)\mathbf{x}'^T(n)\mathbf{c}''}_{D} \\
& + \underbrace{[\mathbf{x}'^T(n)\mathbf{g}''(n)]^2}_{E} - \underbrace{[\mathbf{x}'^T(n)\mathbf{c}'']^2}_{F} \\
& + \underbrace{2\mathbf{x}''^T(n)\mathbf{c}'''\mathbf{x}'^T(n)\mathbf{g}''(n)}_{G} - \underbrace{2\mathbf{x}''^T(n)\mathbf{c}'''\mathbf{x}'^T(n)\mathbf{c}''}_{H}
\end{aligned}
\tag{6.2.6}
$$

Substituting (6.2.6) into (6.1.4) the steady-state fractional tap-length update equation can be rewritten as:

$$l_f(n+1) = l_f(n) - (\alpha + \gamma(A - B + C - D + E - F + G - H)) \tag{6.2.7}$$

where terms A, B, C, D, E, F, G and H are denoted in (6.2.6).

Next a steady-state performance analysis will be given based on this update equation.

6.2.1 Steady-state performance analysis

Before the steady-state analysis, the assumption is made that the system has arrived at steady-state if the quantities $E\{(e_{L(n)}^{(L(n))})^2\}$ and $E\{l_f(n)\}$ tend to constants as $n \to \infty$. To simplify the analysis several further assumptions are also made:

A.1. At steady-state, the tap-length will converge, or can be approximately deemed to have converged to a fixed value $L(\infty)$. As will be shown by the simulations, if all the parameters are set properly, the tap-length will slightly fluctuate around a fixed value. Also the steady-

state tap-length is assumed to be larger than the optimal tap-length $L(\infty) > L_{opt}$. The justification for this assumption is that in most simulations, if the parameter γ is not chosen too small, the steady-state tap-length $L(\infty)$ will be always larger than L_{opt}. This overestimate phenomenon is also justified and discussed in [12], and can be seen in the simulations in the next subsection.

A.2. Both the input signal $x(n)$ and the noise signal $v(n)$ are statistically independent identically distributed (i.i.d) zero mean Gaussian white signals with variances σ_x^2 and σ_v^2, respectively.

A.3. The tail elements of the unknown optimal filter vector $\mathbf{c}_{L_{opt}}$ can be deemed to be drawn from a random white sequence with zero mean and variance σ_c^2. This assumption is used to simplify the analysis. Note that even for a filter with a decaying impulse response structure, the tail elements can be approximately deemed as to have the same variance if the tap-length is long enough, thus this assumption matches the observations in many applications.

A.4. At steady-state, the vectors $\mathbf{g}'(n)$ and $\mathbf{g}''(n)$, which are due to the adaptive noise, are independent of $\mathbf{x}_N(n)$. The justification of this assumption is that the updates of $\mathbf{g}'(n)$ and $\mathbf{g}''(n)$ only depend on the past input vectors, and with assumption *A.2*, $\mathbf{x}_N(n)$ is independent of $\mathbf{x}_N(j)$ if $j \neq n$, thus $\mathbf{g}'(n)$ and $\mathbf{g}''(n)$ are independent of $\mathbf{x}_N(n)$ [64]. Also, in order to simplify the analysis, the following assumption is made that at steady-state

$$E\{\begin{pmatrix} \mathbf{g}'(n) \\ \mathbf{g}''(n) \end{pmatrix} \begin{pmatrix} \mathbf{g}'(n) \\ \mathbf{g}''(n) \end{pmatrix}^T\} = \sigma_g^2 I \tag{6.2.8}$$

where I is the identity matrix and σ_g^2 is the variance of the elements of

$\mathbf{g}'(n)$ and $\mathbf{g}''(n)$.

Also the tap-length is constrained to be not less than a lower floor value L_{min}, where $L_{min} > \Delta$, during its evolution, i.e., if the tap-length fluctuates under L_{min}, it will be set to L_{min}. This operation is necessary since $L(n) - \Delta$ is used as a tap length in the FT algorithm, as can be seen in (6.1.4), and it should be positive.

Taking expectation of both sides of (6.2.7) yields

$$E\{(A - B + C - D + E - F + G - H)\} = -\frac{\alpha}{\gamma} \qquad (6.2.9)$$

Using assumptions *A.1* and *A.3* the following equations are obtained

$$\mathbf{c}''' = \mathbf{0} \qquad (6.2.10)$$

and

$$\|\mathbf{c}''\|_2^2 \approx \begin{cases} (L_{opt} + \Delta - L(\infty))\sigma_c^2 & \text{if } L_{opt} < L(\infty) \leq L_{opt} + \Delta \\ 0 & \text{if } L(\infty) > L_{opt} + \Delta \end{cases} \qquad (6.2.11)$$

With equation (6.2.10) it is straightforward to see that the terms G and H in equation (6.2.9) will disappear at steady-state. Using assumptions *A.1*, *A.2*, *A.3* and *A.4*, the expectations of terms A, B, C and D will be zero at steady-state, and equation (6.2.9) can be written as

$$E\{(A - B + C - D + E - F)\} = \sigma_x^2(E\{\|\mathbf{g}''(n)\|_2^2\} - \|\mathbf{c}''\|_2^2) \qquad (6.2.12)$$

It is straightforward to see that if $L(\infty) > L_{opt} + \Delta$ it will imply that equations (6.2.9) and (6.2.12) contradict each other, i.e., together with

(6.2.11) the r.h.s. of (6.2.12) will be larger than zero if $L(\infty) \gtrsim L_{opt}+\Delta$, but the r.h.s. of (6.2.9) is a negative value. Therefore the conclusion can be made that $L(\infty) \leq L_{opt} + \Delta$, so that by also exploiting assumption *A.1*, the condition that $L_{opt} \leq L(\infty) \leq L_{opt} + \Delta$ will always be used in the following derivations.

In a manner similar to [1], in order to speed up the convergence rate of the FT variable tap-length LMS algorithm, the step size is made variable rather than fixed, according to the range of μ described in [1]:

$$\mu(n) = \mu'/((L(n) + 2)\sigma_x^2) \tag{6.2.13}$$

where μ' is a constant. With this variable step size the term $E\{\|\mathbf{g}''(n)\|_2^2\}$ can be derived as in (6.6.9) in Appendix A. Substituting (6.2.11) and (6.6.9) into (6.2.12) yields

$$E\{A-B+C-D+E-F\} \approx \sigma_x^2 \left[\frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}\sigma_x^2} - (L_{opt} + \Delta - L(\infty))\sigma_c^2\right] \tag{6.2.14}$$

Utilizing (6.2.9) in (6.2.14) yields

$$-\frac{\alpha}{\gamma\sigma_x^2} \approx \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}\sigma_x^2} - (L_{opt} + \Delta - L(\infty))\sigma_c^2 \tag{6.2.15}$$

From equation (6.2.15) the following equation can be obtained

$$L(\infty) \approx L_{opt} + \Delta - \frac{\alpha}{\gamma\sigma_x^2\sigma_c^2} - \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}\sigma_x^2\sigma_c^2} \tag{6.2.16}$$

This equation gives a mathematical formulation of the steady-state tap-length. Since the steady-state tap-length value given in (6.2.16) will seldom be an integer, in practice the steady-state tap-length will fluc-

tuate around this value. This can be clearly seen in later simulations. The steady-state mean square error (MSE) can then be easily obtained with the steady-state tap-length given in (6.2.16) by using the analysis results provided in [1]. In practice, the last two terms of the r.h.s. of equation (6.2.16) will be small, and the steady-state tap-length will be close to the value $L_{opt} + \Delta$, as will be shown in the later simulations.

To avoid the under-modelling situation, the parameters should be chosen to make $L(\infty) > L_{opt}$, and obtain a small fluctuation of the steady-state tap-length. Next some guidelines for the parameter choice will be given.

6.2.2 Guidelines for the parameter choice

In this section some general guidelines for parameter choice are provided. To choose the parameters properly, estimations of the optimal tap-length, the input variance, the noise variance, and the desired system MSE are needed. The availability of these estimations will be application dependent and therefore outside of the scope of this study. With these estimated values, the parameters used in the FT algorithm can be determined as follows:

1. The parameter δ in (6.1.5) is not a crucial parameter, since it is just used to obtain an integer value of the tap-length for the coefficients adaptation, and can be easily set to a small integer.

2. The choice of the parameter μ' can be determined according to the system MSE requirement. Similar to the step size choice of the LMS algorithm, μ' can be a large value in low noise conditions to obtain faster convergence of the MSE, and should be small in high noise conditions to avoid a large MSE.

3. The parameter Δ should be as large as possible to obtain a fast convergence rate of the tap-length, but also much smaller than the estimate of L_{opt}, so that the steady-state tap-length formulated in (6.2.16) will not be significantly biased from L_{opt}. For example, $\Delta \approx 0.1L_{opt}$ will be a good choice for a wide range of optimal tap-lengths.

4. The leakage parameter α should not be too large, so that it will not influence the initial tap-length convergence rate too much. The parameter α should not be too small either, so that once the tap-length is over estimated, α can make the tap-length converge close to the steady-state value as soon as possible. For example, $\alpha = 0.0001$ is not a good choice, since it means that after $10,000$ iterations, the leakage parameter α will reduce the tap-length by one tap, which is usually too slow. Generally, values between 0.001 and 0.01 are good choices for α.

5. The parameter γ is the step size parameter which controls the adaptation process of the variable tap-length. Similar to the step size in the LMS algorithm, a large parameter γ will speed up the convergence rate of the tap-length, but will result in a large fluctuation of the steady-state tap-length. On the other hand, a small parameter γ can obtain a small fluctuation of the steady-state tap-length, but lead to a slow convergence rate. Thus γ provides a trade-off between the convergence rate of the tap-length and the steady-state tap-length variance. The choice of this parameter is important in the FT algorithm. A detailed discussion for the choice of this parameter is as follows:

At first, to avoid under-modelling the optimal tap-length, the steady-state tap-length of the FT method, $L(\infty)$, should not be less than L_{opt}.

Considering the fluctuation of the steady-state tap-length, the parameter γ should be set properly so that $L(\infty) > L_{opt} + \kappa\delta$, where κ is a small positive integer and can be chosen according to the system requirement of the fluctuation of the steady-state tap-length. For example, $\kappa = 2$ is a reasonable choice. Substituting (6.2.16) into the inequality $L(\infty) > L_{opt} + \kappa\delta$ the lower bound value γ_l is obtained:

$$\gamma_l = \frac{\alpha}{(\Delta - \kappa\delta)\sigma_x^2\sigma_c^2 - \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}}} \tag{6.2.17}$$

Secondly, the parameter γ should not be too large to avoid a large fluctuation of the steady-state tap-length. The update process for the steady-state fractional tap-length is formulated in (6.2.7). The fluctuation of the steady-state fractional tap-length is brought about by the fluctuation of the term $\alpha + \gamma(A - B + C - D + E - F + G - H)$. It is straightforward to see in (6.2.7) that large variance of the term $\alpha + \gamma(A - B + C - D + E - F + G - H)$, which is denoted as σ_f^2, will result in large fluctuations of the steady-state fractional tap-length. To avoid such a situation, a simple and intuitive approach is to make the standard deviation σ_f much smaller than the parameter δ in equation (6.1.5), so that the probability of the steady-state fractional tap-length fluctuating outside the range $(L(\infty) - \delta, L(\infty) + \delta)$ can be very small. A simple criterion to satisfy such a requirement is

$$\sigma_f < \rho\delta \tag{6.2.18}$$

where ρ is a small positive value and can be decided according to the system requirement of the fluctuation of the steady-state tap-length. The derivation of the variance σ_f^2 is given in (6.7.18) in Appendix B.

Using (6.7.18) in (6.2.18) and after rearrangement the upper bound value γ_u is obtained:

$$\gamma_u = \frac{-\frac{K_2\alpha}{K_3} + \sqrt{\frac{K_2^2\alpha^2}{K_3^2} + \frac{4(\rho^2\delta^2+\alpha^2)}{K_3}}}{2} \tag{6.2.19}$$

where K_2 and K_3 are respectively formulated in Appendix B (6.7.20) and (6.7.21).

The parameter ρ should be chosen so that the possibility of the tap-length fluctuating under L_{opt} is nearly zero. In general, for high noise condition, this parameter should be chosen small, and for low noise condition it can be chosen larger. Examples for the choice of ρ can be seen in the simulations in the next subsection. With the lower bound value given in (6.2.17) and the upper bound value given in (6.2.19), the parameter γ can then be easily chosen. According to the motivation of the upper bound value γ_u, the values close to this value are good choices to avoid a large fluctuation of the steady-state tap-length while retaining as quick as possible convergence rate; thus, in practice γ should be chosen close to γ_u and larger than γ_l.

Since in practice, all the parameters σ_x^2, σ_v^2, σ_c^2, especially the parameter L_{opt} are unknown, approximate estimations of these parameters can be used in the calculations. Next several simulations will be performed to confirm the above analysis and discussions.

6.2.3 Simulation

In this subsection two simulations are performed to support the analysis and discussions in the previous subsection. In the first simulation a low noise condition is used while a high noise environment is utilized in the

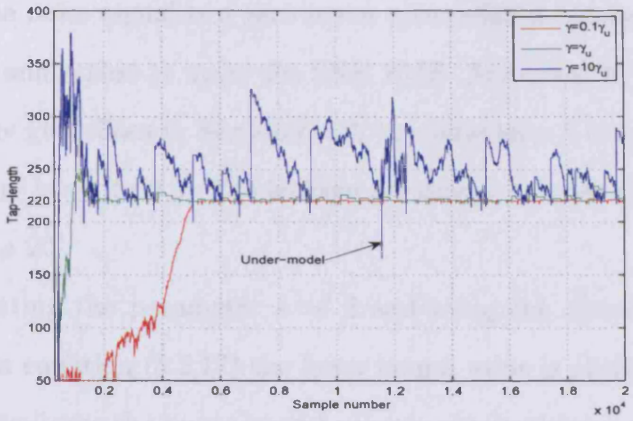

Figure 6.1. The evolution curves of the tap-length with different step sizes under a low noise condition, SNR=20dB.

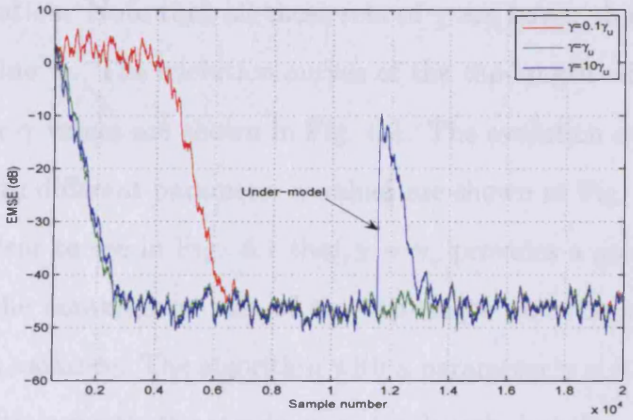

Figure 6.2. The evolution curves of the EMSE with different step sizes under a low noise condition, SNR=20dB.

second simulation.

Low noise case: SNR = 20dB

The setup of this simulation is as follows. The impulse response sequence of the unknown filter is a white Gaussian sequence with zero mean and variance 0.01. The tap-length L_{opt} is set to 200. The input signal is another white Gaussian sequence with zero mean and unit vari-

ance. The noise signal is a zero mean uncorrelated random Gaussian sequence and scaled to make the SNR 20dB. According to the parameter choice guidelines in Section 6.2.2, the parameter δ is set as 2. The step size μ' is set to 0.5. The leakage parameter α is set to 0.005, and Δ is set to 20.

By setting the parameter $\kappa = 2$ and using the above parameter settings in equation (6.2.17) the lower bound value is obtained as $\gamma_l = 0.0314$. Similarly, the upper bound value is obtained as $\gamma_u = 17.954$ by using $\rho = 0.5$ in (6.2.19). To compare the performance with different values of γ, $\gamma = 0.1\gamma_u$, $\gamma = \gamma_u$ and $\gamma = 10\gamma_u$ are respectively used in the simulation. Note that all these sets of γ are larger than the lower bound value γ_l. The evolution curves of the tap-length with different parameter γ values are shown in Fig. 6.1. The evolution curves of the EMSE with different parameter γ values are shown in Fig. 6.2.

It is clear to see in Fig. 6.1 that $\gamma = \gamma_u$ provides a good trade off between the convergence rate of the tap-length and the steady-state tap-length variance. The algorithm with a parameter $\gamma = 0.1\gamma_u$ gives a very smooth curve for the steady-state tap-length, but the convergence rate of both the tap-length and the EMSE is very slow. The algorithm with a parameter $\gamma = 10\gamma_u$ provides a quick convergence rate, but the tap-length fluctuates greatly. When the tap-length is under-modelled, i.e., $L(n) < L_{opt}$, the EMSE will increase, as can be seen in Fig. 6.2.

Substituting all the parameter sets into (6.2.16) the theoretical values of the steady-state tap-length are obtained $L(\infty|\gamma = 0.1\gamma_u) = 219.65$, $L(\infty|\gamma = \gamma_u) = 219.90$ and $L(\infty|\gamma = 10\gamma_u) = 219.93$, which are all close to $L_{opt}+\Delta$. It can be seen from Fig. 6.1 that for the parameter sets $\gamma = \gamma_u$ and $\gamma = 0.1\gamma_u$ the simulation results of the steady-state

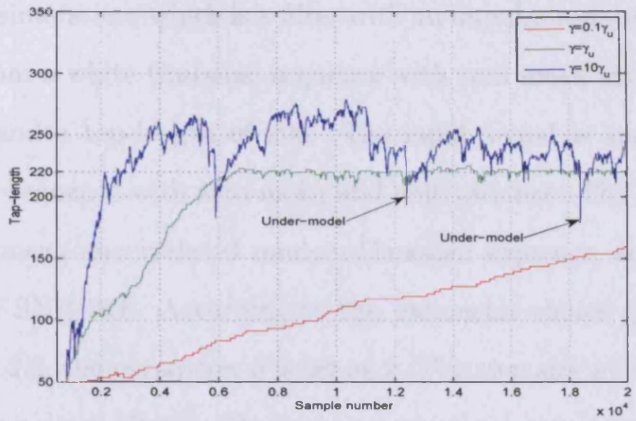

Figure 6.3. The evolution curves of the tap-length with different step sizes under a high noise condition, SNR=0dB.

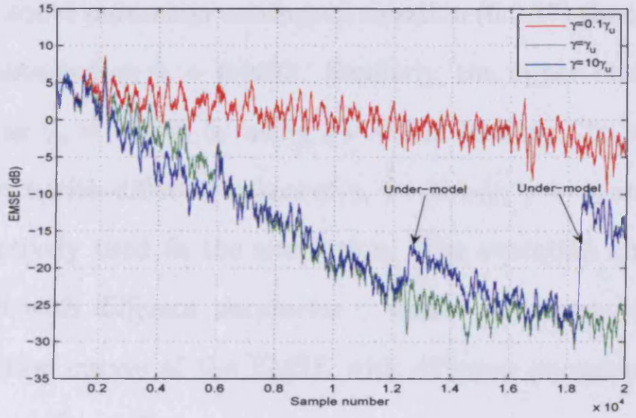

Figure 6.4. The evolution curves of the EMSE with different step sizes under a high noise condition, SNR=0dB.

tap-length match the theoretical values very well: the steady-state tap-length fluctuates around the theoretical value $L(\infty)$, which confirms (6.2.16).

High noise case: SNR = 0dB

In this simulation a high noise environment is used. The setup for this simulation is as follows. The unknown filter is the same as that in the

previous simulation, which is a filter with an impulse response sequence drawn from a white Gaussian sequence with zero mean and a variance of 0.01, and a tap-length of 200. The input signal is another white Gaussian sequence with zero mean and unit variance. The noise signal is a zero mean uncorrelated random Gaussian sequence, and scaled to make the SNR 0dB. According to the parameter choice guidelines in Section 6.2.2, the parameter δ is set as 2. The step size μ' is set as 0.05 to obtain a small EMSE. The leakage parameter α is set to 0.005. Δ is set to 20.

Similar to the first simulation, by setting the parameter $\kappa = 2$, and using the above parameter settings in equation (6.2.17) the lower bound value is obtained as $\gamma_l = 0.0323$. Similarly, the upper bound value is obtained as $\gamma_u = 1.0255$ by using $\rho = 0.2$ in (6.2.19). To compare the performance with different values of γ, $\gamma = 0.1\gamma_u$, $\gamma = \gamma_u$ and $\gamma = 10\gamma_u$ are respectively used in the simulation. The evolution curves of the tap-length with different parameter γ values are shown in Fig. 6.3. The evolution curves of the EMSE with different parameter γ values are shown in Fig. 6.4.

Again from Fig. 6.3 it is clear to see that $\gamma = \gamma_u$ provides a good trade off between the convergence rate of the tap-length and the steady-state tap-length variance. The convergence rate of both the tap-length and EMSE with parameter $\gamma = 0.1\gamma_u$ is too slow for the algorithm. The algorithm with parameter $\gamma = 10\gamma_u$ provides a quick convergence rate of the tap-length, but the fluctuation of the steady-state tap-length is very large. Once the tap-length fluctuates under L_{opt}, EMSE will increase, as can be seen in Fig. 6.4.

Substituting all the parameter sets into (6.2.16) the theoretical val-

ues of the steady-state tap-length are obtained as $L(\infty|\gamma = 0.1\gamma_u) = 214.\dot{6}$, $L(\infty|\gamma = \gamma_u) = 219.0$ and $L(\infty|\gamma = 10\gamma_u) = 219.4$, which are all close to $L_{opt} + \Delta$. For $\gamma = \gamma_u$ the simulation results of the steady-state tap-length match the theoretical values very well, which confirms equation (6.2.16).

To obtain both a fast convergence rate and a small steady-state EMSE for high noise condition, the convex combination approach can be considered, in which two filters are updated simultaneously with different parameters $\gamma = 10\gamma_u$ and $\gamma = \gamma_u$, so that the overall filter can obtain both a rapid convergence rate from the fast filter and a smooth curve for the steady-state tap-length from the slow filter. This convex combination approach will be described in the next section.

6.3 Convex combination approach for the FT algorithm

Although the FT method is more robust and has lower computational complexity compared with other methods [12], its performance can depend on the parameter choice, particularly when the noise level is high. In a high noise environment, SNR$\leq$0dB, fixed parameters which achieve both fast convergence rate and small steady-state variance of the tap-length will be difficult to obtain, as can be seen in the previous section. As described in [65] and [66] the convex combination of adaptive filters can improve the performance of adaptive schemes. A convex combination of adaptive filters is introduced in this section to improve the performance of the FT method in high noise environments. Simulations will be performed to support the advantages of this new approach.

6.3.1 Convex combination of adaptive filters

In previous research into a convex combination structure, two filters are adapted separately [65]. The output signals and the output errors of both filters are combined in such a manner that the advantages of both component filters are retained: the rapid convergence from the fast filter and the reduced steady-state error from the slow filter. Here the first filter is assumed to have a fast convergence rate throughout this section. The output of the overall filter is

$$y(n) = \lambda(n)y_1(n) + (1 - \lambda(n))y_2(n) \tag{6.3.1}$$

where $y_i(n) = \mathbf{w}_i^T(n)\mathbf{x}_i(n)$, $i = 1, 2$, $\mathbf{w}_i(n)$ and $\mathbf{x}_i(n)$ are the adaptive filter weight vector and input vector of the ith filter, and $\lambda(n) \in [0, 1]$ is a mixing scalar parameter. The output error of the overall filter is

$$e(n) = \lambda(n)e_1(n) + (1 - \lambda(n))e_2(n) \tag{6.3.2}$$

where $e_1(n)$ and $e_2(n)$ are the output errors of the two component filters:

$$e_i(n) = d(n) - \mathbf{w}_i^T(n)\mathbf{x}_i(n), i = 1, 2 \tag{6.3.3}$$

The key point of the convex combination of adaptive filters is to control the overall filter by the mixing parameter $\lambda(n)$ according to the performance of the two component filters.

In the convex combination of adaptive filters, the mixing parameter $\lambda(n)$ is adapted to minimize the quadratic error of the overall filter [65]. Rather than adapting $\lambda(n)$ directly, a variable parameter $a(n)$ which defines $\lambda(n)$ via a sigmoidal function is adopted. The sigmoidal function

is

$$\lambda(n) = sgm(a(n)) = (1 + e^{-a(n)})^{-1} \tag{6.3.4}$$

and the update equation for $a(n)$ is given by

$$a(n+1) = a(n) + \mu_a e(n)[y_1(n) - y_2(n)]\lambda(n)[1 - \lambda(n)] \tag{6.3.5}$$

where μ_a is the step size of the adaptation of $a(n)$ and should be chosen appropriately to obtain a fast and stable convergence of the combination. The parameter $a(n)$ is also restricted to the interval $[-a^+, a^+]$ which limits the permissible range of $\lambda(n)$ to $[1 - \lambda+, \lambda+]$ where $\lambda+ = sgm(a^+)$ is a constant close to unity [65]. A good choice for a^+ is 4, which constrains $\lambda(n)$ to $[0.018, 0.982]$. This value has been used in [65] and also will be used in the later simulations.

As shown in [65] the steady-state performance of the convex combination of adaptive filters is better or as close as desired to its best component filter. By denoting the noise contained in the desired signal $d(n)$ as $v(n)$, and defining the EMSE of the overall filter as $J_{ex}(n) = E\{(e(n) - v(n))^2\}$, the advantage of the convex combination structure can be shown as [65]:

$$J_{ex}(\infty) \leq \min[J_{ex,1}(\infty), J_{ex,2}(\infty)] \tag{6.3.6}$$

where $J_{ex}(\infty)$, $J_{ex,1}(\infty)$ and $J_{ex,2}(\infty)$ denote respectively the steady-state EMSE of the overall filter, the first component filter and the second component filter. Note that no assumption is made about the specific nature of the adaptive filter, and thus (6.3.6) is suitable for any adaptive algorithm [65].

Two modifications have been proposed to improve the performance of the original convex combination algorithm [66], and both are used in the later simulations. One of the modifications is to take advantage of the fast filter to speed up the convergence of the slow one. It is achieved by modifying the adaptation of the slow component filter at an early stage of the adaptation to approach that of the fast component filter.

Another modification is to improve the convergence of the parameter $a(n)$. It is clear that when both outputs of the two filters are similar, the factor $y_1(n)-y_2(n)$ in (6.3.5) will be very small and the convergence of $a(n)$ will be slow. A momentum term for adapting parameter $a(n)$ is then added to alleviate this problem [66]:

$$a(n+1) = a(n) + \mu_a e(n)[y_1(n) - y_2(n)]\lambda(n)[1-\lambda(n)] + \rho(a(n) - a(n-1)) \quad (6.3.7)$$

where ρ is a positive constant. In general, 0.5 is a good choice of the parameter ρ, as shown in [66]. Compared with the basic convex combination of adaptive filters these modifications have improved the convergence rate of the overall filter. Next this convex combination of filters will be applied in the FT variable tap-length algorithm.

6.3.2 Convex combination filters for the FT algorithm

In the structure of convex combination for the FT algorithm, two component adaptive filters are utilized to implement the FT method separately, but the philosophy for selecting parameters γ, α and Δ in (6.1.4) of each component filter is different. The parameters in the first filter are set to provide quick convergence rate of the tap length, whereas

parameters for the second filter are set to provide a smooth curve of the evolution of the tap-length, which results in a small steady-state EMSE, as have been discussed in the previous section.

Both modifications of the basic convex combination structure which have been introduced in the previous subsection are adopted in the simulations. In the first modification, a simple criterion is set up to decide how and when should the modification be adopted for the adaptation of the fractional tap-length of the slow component filter:

$$l_{f2}(n+1) = \phi l_{f2}(n) + (1-\phi) l_{f1}(n), \text{if } \lambda(n) > t \tag{6.3.8}$$

where l_{fi} denotes the fractional tap-length of the ith filter, t is a threshold between 0.5 and 1, and ϕ is a weight parameter close to but less than unity. This criterion is easy to understand: when $\lambda(n)$ is larger than t, the first filter performs better than the second filter, and the fractional tap-length of the second filter should be modified to approach to that of the first filter, to speed up the convergence rate of the second filter. In general, a value between 0.6 and 0.8 is a good choice for t, and the value 0.99 is a good choice for ϕ, which gives both fast and stable convergence for the fractional tap-length of the second filter. The second modification is described by (6.3.7).

By denoting the initial tap lengths of both component filters as L_{ini}, the step size for the adaptation of the weights vector of both component filters as μ, the implementation of this convex combination method can then be summarized as follows:

Initialization: α_1, α_2, γ_1, γ_2, Δ_1, Δ_2, μ, μ_a, δ, ρ, t, ϕ, L_{ini}, a^+. All these values should be set according to the system requirement,

and an example of the set of these parameters can be seen in the later simulations.

Update at each iteration:

1. Update the filter coefficients of both component filters by using the LMS or related algorithm with the current tap-lengths.

2. Adapt the fractional tap-lengths of both component filters respectively according to (6.1.4);

3. Calculate parameters $a(n)$ and $\lambda(n)$ according to (6.3.4) and (6.3.7);

4. Modify the fractional tap-length of the slow component filter according to (6.3.8).

5. Update the current tap-lengths of both filters according to (6.1.5).

As will be confirmed in the later simulations, the first component filter provides a good tracking ability of the tap-length for the overall filter. On the other hand, a smooth curve of tap-length and small EMSE is obtained from the second component filter. Through appropriate update of the mixing parameter λ, both advantages of these two filters are extracted and a better performance can thereby be obtained. Similarly to (6.3.6), the overall EMSE performance of the convex combination of filters is better or as close as desired to the best component filter.

6.3.3 Simulation

A simulation is provided in this subsection to support the advantages of the proposed convex combination approach. The setup of this simulation is similar to that in [12]. In the simulations in [12] a low noise environment where SNR is 20dB is used. Since the performance of the

proposed approach is comparable with the FT method in à low noise level, a high noise level environment where the SNR is 0dB is used in the simulation, to highlight the advantages of the proposed approach. The normalized LMS (NLMS) algorithm [15] [16] is also used in this simulation.

The input signal $x(n)$ is obtained by passing white Gaussian noise through a spectral shaping filter with a z-domain transfer function of $H(z) = 0.35 + z^{-1} + 0.35z^{-2}$. Similar as that in [12], two unknown systems are tested:

$$\mathbf{h}_1 = \sum_{k=1}^{80} a_k z^{-k}, \mathbf{h}_2 = \sum_{k=1}^{30} b_k z^{-k} \tag{6.3.9}$$

where a_k and b_k are chosen from a white Gaussian random sequence with zero mean and unit variance. The desired signal is obtained by filtering the input signal with $\mathbf{h}_1$ or $\mathbf{h}_2$ in different time intervals:

$$d(n) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{6.3.10}$$

where $\mathbf{w}(n) = \mathbf{h}_1$ for $n < 10,000$ or $n \geq 20,000$ and $\mathbf{w}(n) = \mathbf{h}_2$ for $10,000 \leq n < 20,000$.

Independent, zero mean Gaussian noise is then added to the unknown system output to provide an SNR of 0dB. The common parameters are set the same for all the following experiments: $\mu = 0.1$, $\mu_a = 1$, $\delta = 2$, $\rho = 0.5$, $t = 0.8$, $\phi = 0.99$, $L_{ini} = 20$, and $a^+ = 4$. Also the tap-lengths during the adaptation are constrained to be no less than L_{ini}. Two sets of parameters are tested with the FT method and the convex combination approach:

A. $\alpha = 0.08$, $\gamma = 4$, $\Delta = 15$.

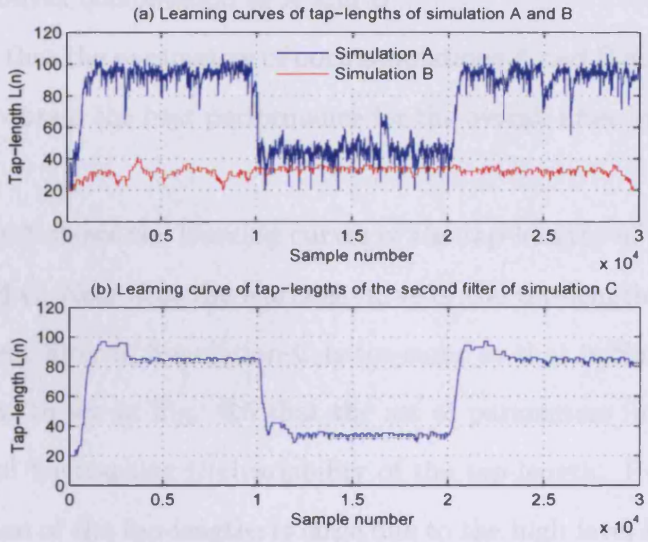

Figure 6.5. Learning curves of tap-lengths of simulations A, B and C

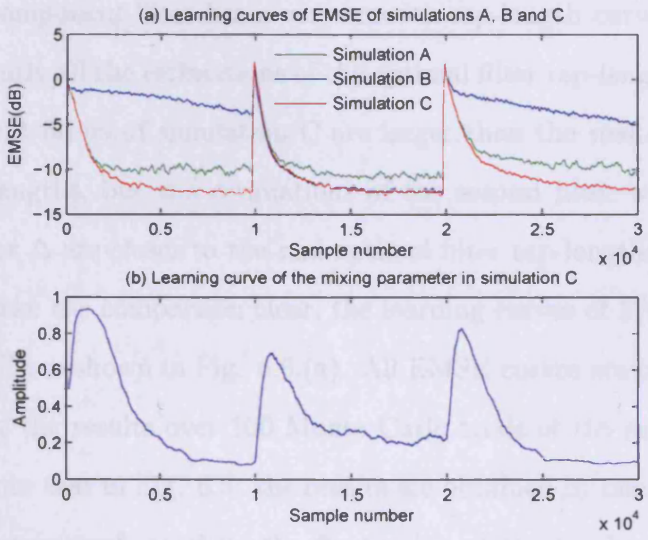

Figure 6.6. Learning curves of EMSE of all simulations and the mixing parameter in Simulation C

B. $\alpha = 0.01$, $\gamma = 0.5$, $\Delta = 4$.

C. Convex combination of A and B

Note that the parameters of both simulations A and B are set empirically to obtain the best performance for the overall filter in simulation C.

Fig. 6.5 shows the learning curves of the tap-lengths of simulations A, B and C. Note that the learning curve of the tap-length of the first component filter in Simulation C is the same as that in Simulation A. It is clear to see in Fig. 6.5 that the set of parameters in simulation A is good for tracking the variability of the tap-length. However, the fluctuation of the tap-lengths is large due to the high level interference signal. The parameter set of simulation B is good for the interval where the optimal tap-length is 30. However, it is too small to estimate the channel length during intervals with optimal tap-length of 80. Both component filters in simulation C have good tracking abilities, and the second component filter has a very smooth tap-length curve. Furthermore, nearly all the estimations of the optimal filter tap-lengths in both component filters of simulation C are larger than the real optimal filter tap-lengths, but the estimations of the second filter with smaller parameter Δ are closer to the real optimal filter tap-lengths.

To make the comparison clear, the learning curves of EMSE rather than MSE are shown in Fig. 6.6.(a). All EMSE curves are obtained by averaging the results over 100 Monte Carlo trials of the same experiment. Note that in Fig. 6.5. the results are obtained by one realization for the experiment, to show the fluctuation of the tap-lengths in the filter with large parameters. It is clear to see in Fig. 6.6.(a) that the EMSE of simulation B is large at the intervals where the optimal filter

length is 80, because the set of its parameters can not estimate the associated tap-length. The EMSE of simulation A is also large over the interval where the optimal filter length is 30 because of the large fluctuations in the tap-length estimate, which may result in under-modelling of the tap-length. Simulation C performs better than both simulations A and B due to the combination approach, and is robust to system variation.

The evolution of the mixing parameter $\lambda(n)$ in simulation C is shown in Fig. 6.6.(b). This curve is also obtained over 100 Monte Carlo trials of the same experiment. From this figure it is clear to see that the parameter $\lambda(n)$ increases towards unity initially, to provide a good tracking performance for the overall filter, and then converges to a small value to obtain a small steady-state EMSE from the second component filter. This behavior is repeated in the different regions of the figure. The evolution of this parameter clearly matches the requirements of the convex combination.

6.4 A new VTLMS algorithm to model an exponential decay impulse response

In many applications such as echo cancelling, the unknown filter exhibits a constant exponential decay envelope. Modelling the unknown impulse response in such applications is typically achieved with a length N FIR filter, denoted by $\mathbf{c}_N$. In practice, N is chosen as a compromise between modelling the significant energy within the impulse response and limiting computational complexity. In this section an adaptive solution is proposed for the choice of N. The evolution of the tap-length of the proposed algorithm is designed in an iterative way to minimize

the MSD at each iteration, which is defined as $E\{\|\mathbf{c}_N - \mathbf{w}_N(n)\|_2^2\}$, where $\mathbf{w}_N(n)$ is the adaptive filter weight vector. The target of the proposed approach is not only to find a good choice of the steady-state tap-length for the adaptive filter, but also to ensure well behaved transient tap length convergence, so that a better performance as compared with the fixed tap-length algorithm is obtained.

In a previous research study [1] a theoretically optimal variable tap-length sequence for the LMS algorithm in such applications has been introduced. However, this algorithm suffers from heavy computational complexity due to solving for Lambert's W-function [1], thus it is not suitable in practice. The derivation of the proposed algorithm is based on the method in [1], but the computational complexity is greatly reduced. As will be shown by the simulation results the proposed algorithm converges faster than the fixed tap-length LMS algorithm, and is very robust to the initial tap-length choice.

6.4.1 The new VTLMS algorithm

For convenience the LMS algorithm is formulated with a system identification model, and the desired unknown filter impulse response sequence is assumed to have a constant exponential decay envelope. In this model the desired signal $d(n)$ is formulated as follows

$$d(n) = \mathbf{x}_N^T(n)\mathbf{c}_N + v(n) \tag{6.4.1}$$

where $(\cdot)^T$ denotes the transpose operator, $\mathbf{x}_N$ is the input vector with a tap length of N, $v(n)$ is the noise signal and $\mathbf{c}_N$ can be modelled as

follows

$$c(i) = e^{(-(i-1)\tau)} r(i), \quad i = 1, ..., N \tag{6.4.2}$$

where $c(i)$ is the ith coefficient of the unknown filter vector $\mathbf{c}_N$, τ is a positive constant to model the decay rate, and $r(i)$ is drawn from a zero mean unit variance Gaussian sequence.

Since in the variable-tap length LMS algorithm, the tap-length is time-varying rather than fixed, the parameter $L(n)$ is used to denote the integer tap-length which is used for the coefficient update of the LMS algorithm at the nth iteration, and assume $L(n) \leq N$. The filter coefficients can be updated as

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e(n) \mathbf{x}_{L(n)}(n) \tag{6.4.3}$$

where $\mathbf{w}_{L(n)}$ and $\mathbf{x}_{L(n)}$ are respectively the $L(n)$-tap adaptive filter vector and the input vector, μ is the step size for the update of the coefficients, and $e(n)$ is the output error defined as

$$e(n) = d(n) - \mathbf{x}^T_{L(n)}(n) \mathbf{w}_{L(n)}(n) \tag{6.4.4}$$

Similar to the formulation in [1], $\mathbf{c}_N$ is split into two parts as

$$\begin{pmatrix} \mathbf{c}' \\ \mathbf{c}'' \end{pmatrix}$$

where $\mathbf{c}'$ is the part modelled by $\mathbf{w}_{L(n)}(n)$ and $\mathbf{c}''$ is the part undermodelled. Defining $\mathbf{g}_N(n)$ as the total coefficient error vector $\mathbf{c}_N - \mathbf{w}_N(n)$ where $\mathbf{w}_N(n)$ is obtained by padding $\mathbf{w}_{L(n)}(n)$ with zeros, the MSD can then be formulated by $E\{\|\mathbf{g}_N(n)\|_2^2\}$.

Since the tap-length should not be constrained to be an integer to find a continuous update, similar to as in [1] and [12] the fractional-tap length concept is used in the following derivations, where the fractional-tap length denoted by $l_f(n)$ will be used in the update of the tap-length, and the tap-length $L(n)$ which is used for the update of the adaptive filter coefficients is assigned to the integer immediately below $l_f(n)$.

Similar to [1], both the input signal $x(n)$ and the noise signal $v(n)$ are assumed to be statistically independent identically distributed (i.i.d) zero mean Gaussian white noise signals with variances σ_x^2 and σ_v^2 respectively. According to the analysis in [1] the evolution of the MSD can be formulated as follows:

$$E\{\|\mathbf{g}_N(n+1)\|_2^2\} = \beta E\{\|\mathbf{g}_N(n)\|_2^2\} + (\eta - \beta)\,\|\mathbf{c}''\|_2^2 + \gamma \tag{6.4.5}$$

where

$$\beta = 1 - 2\mu\sigma_x^2 + (l_f(n+1) + 2)\mu^2\sigma_x^4, \tag{6.4.6}$$

$$\eta = 1 + l_f(n+1)\mu^2\sigma_x^4 \tag{6.4.7}$$

and

$$\gamma = l_f(n+1)\mu^2\sigma_x^2\sigma_v^2 \tag{6.4.8}$$

The range of the step size which ensures the convergence of (6.4.5) is [1]

$$0 < \mu < \frac{2}{(l_f(n+1)+2)\sigma_x^2} \tag{6.4.9}$$

At first to speed up the convergence rate of the LMS algorithm, the step size is made variable rather than fixed, according to the range of

μ described in (6.4.9). In [1] the step size is set to $\mu(n) = \mu'/((l_f(n) + 2)\sigma_x^2)$ where μ' is a fixed constant and less than 2. To remove the dependence between the step size $\mu(n)$ and the fractional tap length $l_f(n)$, and noting that $l_f(n-1)$ is very close to $l_f(n)$, the step size is set as follows

$$\mu(n) = \mu'/((l_f(n-1) + \delta)\sigma_x^2) \tag{6.4.10}$$

where δ is an integer larger than 2 to ensure stability of the algorithm.

Secondly, the squared norm of the partial response of $\mathbf{c}_N$ can be expressed in the form [1]

$$\|\mathbf{c}''\|_2^2 = \frac{e^{-2l_f(n)\tau} - e^{-2N\tau}}{1 - e^{-2N\tau}} \|\mathbf{c}_N\|_2^2 \tag{6.4.11}$$

As shown in [1] the derivative $(\partial E\{\|\mathbf{g}_N(n+1)\|_2^2\}/\partial l_f^2(n+1))$ is positive, thus a tap-length $l_f(n+1)$ exists to minimize the term $E\{\|\mathbf{g}_N(n+1)\|_2^2\}$. Replacing μ in (6.4.5) with $\mu(n+1)$ and substituting (6.4.6), (6.4.7), (6.4.8) and (6.4.11) into (6.4.5), and setting $(\partial E\{\|\mathbf{g}_N(n+1)\|_2^2\}/\partial l_f(n+1)) = 0$ yields

$$\mu(n+1)\sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} - 4\tau(1 - \mu(n+1)\sigma_x^2)\frac{e^{-2l_f(n+1)\tau}}{1 - e^{-2N\tau}} \|\mathbf{c}_N\|_2^2 + l_f(n+1)\mu(n+1)\sigma_v^2 = 0 \tag{6.4.12}$$

After rearranging the following equation can be obtained

$$l_f(n+1) = -\frac{1}{2\tau} \log \frac{\mu(n+1)\sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} + \mu(n+1)\sigma_v^2}{4\tau(1 - \mu(n+1)\sigma_x^2)\frac{\|\mathbf{c}_N\|_2^2}{1-e^{-2N\tau}}} \tag{6.4.13}$$

Substituting (6.4.10) into (6.4.13) yields

$$l_f(n+1) = -\frac{1}{2\tau}\log\frac{\mu'\sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} + \mu'\sigma_v^2}{4\tau\sigma_x^2(l_f(n)+\delta-\mu')\frac{\|\mathbf{c}_N\|_2^2}{1-e^{-2N\tau}}} \tag{6.4.14}$$

By defining $\Delta l_f = l_f(n+1) - l_f(n)$ and using equation (6.4.14) the following equation is obtained

$$\Delta l_f = -\frac{1}{2\tau}\log\frac{(\sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} + \sigma_v^2)(l_f(n-1)+\delta-\mu')}{(\sigma_x^2 E\{\|\mathbf{g}_N(n-1)\|_2^2\} + \sigma_v^2)(l_f(n)+\delta-\mu')} \tag{6.4.15}$$

The update of the tap-length of the new variable tap-length LMS algorithm is then obtained:

$$l_f(n+1) = l_f(n) - \frac{1}{2\tau}\log\frac{(\sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} + \sigma_v^2)(l_f(n-1)+\delta-\mu')}{(\sigma_x^2 E\{\|\mathbf{g}_N(n-1)\|_2^2\} + \sigma_v^2)(l_f(n)+\delta-\mu')} \tag{6.4.16}$$

Assuming that the input signals are independent of the adaptive weight coefficients the following equation exists [15]

$$E\{e^2(n)\} = \sigma_x^2 E\{\|\mathbf{g}_N(n)\|_2^2\} + \sigma_v^2 \tag{6.4.17}$$

Substituting (6.4.17) into (6.4.16) yields

$$l_f(n+1) = l_f(n) - \frac{1}{2\tau}\log\frac{E\{e^2(n)\}(l_f(n-1)+\delta-\mu')}{E\{e^2(n-1)\}(l_f(n)+\delta-\mu')} \tag{6.4.18}$$

In practice the statistical average term $E\{e^2(n)\}$ can be approxi-

mated by it's time average estimation $\overline{e^2(n)}$, which can be obtained as:

$$\overline{e^2(n)} = \phi\overline{e^2(n-1)} + (1-\phi)e^2(n) \tag{6.4.19}$$

where ϕ is a positive constant close to but less than unity. The new variable-tap length algorithm is then obtained as follows:

$$l_f(n+1) = l_f(n) - \frac{1}{2\tau}\log\frac{\overline{e^2(n)}(l_f(n-1)+\delta-\mu')}{\overline{e^2(n-1)}(l_f(n)+\delta-\mu')} \tag{6.4.20}$$

Since the tap length which is used in the update of the filter coefficients must be an integer, the floor of $l_f(n+1)$ is chosen for the coefficient update:

$$L(n+1) = \lfloor l_f(n+1) \rfloor \tag{6.4.21}$$

where $\lfloor . \rfloor$ is the floor operator which rounds down the embraced value to the nearest integer. By replacing the μ in equation (6.4.3) with $\mu(n)$, the full adaptive algorithm can consequently then be implemented by equations (6.4.4), (6.4.10), (6.4.3), (6.4.19), (6.4.20) and (6.4.21).

6.4.2 Steady-state performance of the proposed algorithm

According to the update equation (6.4.20) it is straightforward to obtain that

$$l_f(\infty) = l_f(0) - \frac{1}{2\tau}\log\frac{\overline{e^2(\infty)}(l_f(0)+\delta-\mu')}{\overline{e^2(0)}(l_f(\infty)+\delta-\mu')} \tag{6.4.22}$$

where $l_f(0)$ and $l_f(\infty)$ are the initial and steady-state values of the fractional tap-length, $\overline{e^2(0)}$ and $\overline{e^2(\infty)}$ are the initial and steady-state values of the smoothed square error. The initial value $\overline{e^2(0)}$ can be set as $\sigma_d^2 + \sigma_v^2$, where σ_d^2 is the variance of the desired signal and can be

formulated as $\sigma_x^2 \|\mathbf{c}_N\|_2^2$. Substituting (6.4.17) into (6.4.22) yields

$$l_f(\infty) = l_f(0) - \frac{1}{2\tau} \log \frac{(\sigma_x^2 E\{\|\mathbf{g}_N(\infty)\|_2^2\} + \sigma_v^2)(l_f(0) + \delta - \mu')}{(\sigma_x^2 \|\mathbf{c}_N\|_2^2 + \sigma_v^2)(l_f(\infty) + \delta - \mu')} \tag{6.4.23}$$

From equation (6.4.5), (6.4.6), (6.4.7), (6.4.8) and (6.4.10) the following equation can be obtained

$$E\{\|\mathbf{g}_N(\infty)\|_2^2\} = \frac{2\sigma_x^2(l_f(\infty) + 2 - \mu') \|\mathbf{c}''\|_2^2 + \mu' l_f(\infty)\sigma_v^2}{(2 - \mu')(l_f(\infty) + 2)\sigma_x^2} \tag{6.4.24}$$

To simplify the formulation, N is assumed to be very large and $l_f(\infty)$ is close to N, thus $l_f(\infty) \approx l_f(\infty) + 2 \approx l_f(\infty) + \delta - \mu'$ and the term $\|\mathbf{c}''\|_2^2$ is very small. Furthermore, by assuming that μ' has been chosen properly so that the term $2\sigma_x^2 \|\mathbf{c}''\|_2^2 \ll \mu'\sigma_v^2$, the following equation can be obtained

$$E\{\|\mathbf{g}_N(\infty)\|_2^2\} \approx \frac{\mu'\sigma_v^2}{(2 - \mu')\sigma_x^2} \tag{6.4.25}$$

Substituting (6.4.25) into (6.4.23) yields

$$l_f(\infty) \approx l_f(0) - \frac{1}{2\tau} \log \frac{2\sigma_v^2(l_f(0) + \delta - \mu')}{(\sigma_x^2 \|\mathbf{c}_N\|_2^2 + \sigma_v^2)(l_f(\infty) + \delta - \mu')(2 - \mu')} \tag{6.4.26}$$

From (6.4.26) the steady-state tap-length $l_f(\infty)$ is correlated with three parameters: the step size μ', the parameter δ and the initial tap-length $l_f(0)$. Assuming that μ' has been chosen properly, and both $L(\infty)$ and $L(0)$ are much larger than μ' and δ, then the influence of δ

and μ' can be ignored. Next a simulation will be performed to show that with wide range in the choice of the initial tap-length $L(0)$, the steady-state tap-length $L(\infty)$ can converge to values which provide a good compromise between modelling the significant energy within the impulse response and limiting computational complexity.

6.4.3 Simulation

In this section the above derivations will be examined, and the proposed algorithm will be compared with the fixed-tap length LMS algorithm and the optimal variable tap length LMS algorithm [1] by simulations. The setup of all the simulations is similar to that in [1]: the unknown filter is a white Gaussian noise sequence with zero mean and a variance of 0.01 weighted by an exponential decay envelope. The tap length is set to 1024, and the decay parameter τ is set to 0.005. One representation of the unknown filter can be seen in Fig. 6.7. The input signal is another white Gaussian noise sequence with zero mean and unit variance. The noise signal is a zero mean random Gaussian sequence with a variance of 0.01. The parameter δ for the proposed algorithm is set to 5. The smoothing parameter ϕ in (6.4.19) is set to 0.99. The step size μ' for both the proposed algorithm and the optimal variable tap-length algorithm is set to 0.5. The initial value of $\overline{e^2(n)}$, i.e., $\overline{e^2(0)}$, is set to $\sigma_d^2 + \sigma_v^2$.

The steady-state MSD with different steady-state tap-length values is shown in Fig. 6.8 (a), calculated from (6.4.24). It is clear to see in Fig. 6.8 (a) that the steady-state MSD decreases with the increase of the steady-state tap-length. However, due to the exponential damping envelope structure as shown in Fig. 1, the MSD will nearly be a

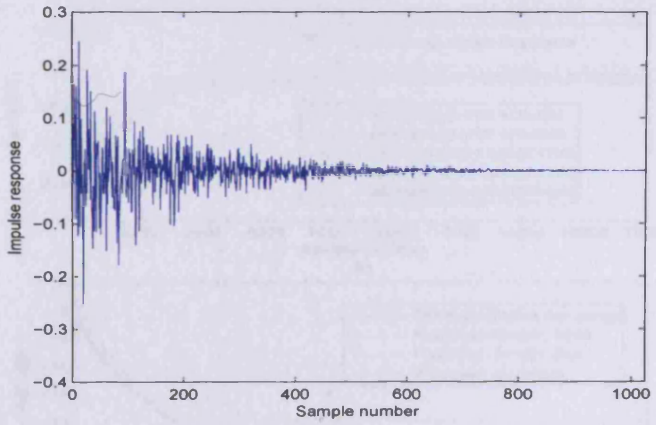

Figure 6.7. One representation of the unknown impulse response sequence

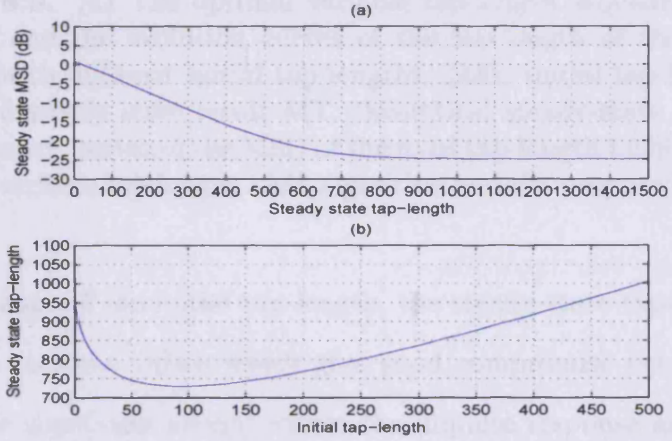

Figure 6.8. (a) Steady-state MSD with different values of the steady-state tap-length according to (6.4.24). (b) Steady-state tap-lengths with different initial tap-length values according to (6.4.26)

constant if the steady-state tap-length is larger than some value, such as 800 in the simulation. The main energy of the unknown impulse response is contained in approximately the first 800 coefficients, which is the part to be found by the proposed approach.

The values of the steady-state tap-length with different initial tap-length values are shown in Fig. 6.8 (b), calculated from (6.4.26). It is clear to see from Fig. 6.8 (b), together with Fig. 6.8 (a) that with

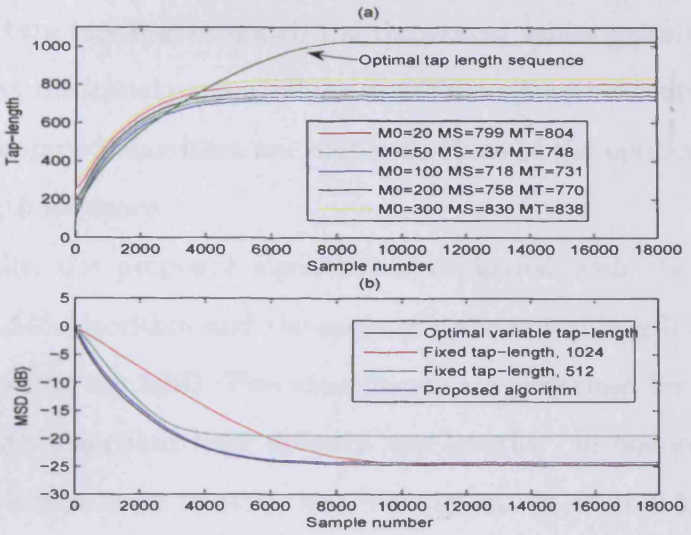

Figure 6.9. (a) The optimal variable tap-length sequence obtained from [1] and the evolution curves of the tap-length of the proposed method with different initial tap-lengths. (M0: initial tap length MS: simulated steady-state result MT: theoretical steady-state result) (b) The evolution curves of the MSD of the fixed tap-length LMS algorithm, optimal variable tap-length LMS algorithm and the proposed algorithm

a wide range of the initial tap-length, the steady-state tap-length can converge to some values which give good compromise between modelling the significant energy within the impulse response and limiting computational complexity, thus the conclusion can be made that the proposed algorithm is robust to the choice of the initial tap-length.

To confirm (6.4.26), several simulations are performed for the proposed algorithm with different initial tap-length values. The evolution curves of the tap-length with different initial tap-length values are shown in Fig. 6.9 (a), where different initial tap-length values, simulated steady-state tap-length values, and the theoretical steady-state tap-length values which are obtained from (6.4.26) are given in the legend of the plot. As a comparison, the optimal variable tap-length is also given. From these values it is clear to see that the simulated

steady-state tap-lengths match the theoretical values quite well. Furthermore, the initial parts of all the variable tap-length evolution curves of the proposed algorithm are similar to those of the optimal variable tap-length sequence.

Finally, the proposed algorithm is compared with the fixed-tap length LMS algorithm and the optimal variable tap length algorithm by comparing the MSD. Two experiments are performed for the fixed-tap length algorithm with different tap lengths. In one experiment the tap length is set to 1024, and the step size is set to 0.5/1024. In another experiment the tap length is set to 512, and the step size is set to 0.5/512. The initial tap length of the proposed algorithm is set to 20. The evolution curves of the MSD of the fixed tap-length LMS algorithm, the optimal variable tap-length LMS algorithm and the proposed algorithm are shown in Fig. 6.9 (b). All the results in Fig. 6.9 are obtained by averaging the results over 100 Monte Carlo trials of the same experiment.

Fig. 6.9 (b) shows that although the steady-state tap-length of the proposed algorithm is less than that of the optimal variable tap-length algorithm, their MSD evolution curves are nearly the same. It is clear to see that with a similar steady-state MSD, the proposed algorithm converges faster than the fixed tap-length LMS algorithm with a tap length of 1024. The convergence rate of the fixed tap length LMS algorithm with a tap length of 512 is fast, but the MSD is large. The proposed algorithm has both fast convergence rate and small MSD, and a good steady-state tap-length is also found in an adaptive way.

6.5 Conclusion

In this chapter, a detailed introduction of a VTLMS algorithm, the FT algorithm has been provided. A steady-state performance analysis for the FT algorithm is given. To improve the performance of the FT algorithm in high noise conditions, a convex combination approach for the FT algorithm is proposed. Furthermore, a new practical VTLMS algorithm is also designed for applications in which the optimal filter has an exponential decay impulse response. All these analyses or new approaches have been confirmed or supported by simulations. The research results in this chapter provide deeper understanding of the VTLMS algorithm. As a developing topic, more research is required for the VTLMS algorithms to make them to be robust to different environments, such as the environment in Chapter 4, where both the input and noise are highly nonstationary speech signals.

6.6 Appendix A: Derivation of the term $E\{\|\mathbf{g}''(n)\|_2^2\}$

As analyzed in [1], with a fixed tap-length, the steady-state value of $E\{\|\mathbf{g}_N(n)\|_2^2\}$ can be expressed as

$$E\{\|\mathbf{g}_N(n)\|_2^2\} = \frac{(s-r)\|\mathbf{c}'''\|_2^2 + t}{1-r} \tag{6.6.1}$$

where

$$r = 1 - 2\mu\sigma_x^2 + (L(\infty)+2)\mu^2\sigma_x^4, \tag{6.6.2}$$

$$s = 1 + L(\infty)\mu^2\sigma_x^4 \tag{6.6.3}$$

and

$$t = L(\infty)\mu^2\sigma_x^2\sigma_v^2 \tag{6.6.4}$$

Moreover, the MSD can be divided into three parts [1]:

$$E\{\|\mathbf{g}_N(n)\|_2^2\} = E\{\|\mathbf{g}'(n)\|_2^2\} + E\{\|\mathbf{g}''(n)\|_2^2\} + \|\mathbf{c}'''\|_2^2 \tag{6.6.5}$$

Substituting (6.6.1) into (6.6.5) yields

$$E\{\|\mathbf{g}'(n)\|_2^2\} + E\{\|\mathbf{g}''(n)\|_2^2\} = \frac{(s-1)\|\mathbf{c}'''\|_2^2 + t}{1-r} \tag{6.6.6}$$

Substituting (6.2.10), (6.6.2), (6.6.3) and (6.6.4) into (6.6.6), and with the approximation $L(\infty) \approx L(\infty) + 2$ the following equation can be obtained

$$E\{\|\mathbf{g}'(n)\|_2^2\} + E\{\|\mathbf{g}''(n)\|_2^2\} \approx \frac{\mu'\sigma_v^2}{(2-\mu')\sigma_x^2} \tag{6.6.7}$$

Using assumption *A.4* results in $E\{\|\mathbf{g}'(n)\|_2^2\} = (L(\infty) - \Delta)\sigma_g^2$ and $E\{\|\mathbf{g}''(n)\|_2^2\} = \Delta\sigma_g^2$, together with equation (6.6.7) the following equation can be obtained

$$E\{\|\mathbf{g}''(n)\|_2^2\} \approx \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L(\infty)\sigma_x^2} \tag{6.6.8}$$

Since $L(\infty) \leq L_{opt} + \Delta$, and in practice $\Delta \ll L_{opt}$, thus if $L(\infty) > L_{opt}$, $L(\infty)$ will be very close to L_{opt}. Equation (6.6.8) can then be approximately written as

$$E\{\|\mathbf{g}''(n)\|_2^2\} \approx \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}\sigma_x^2} \tag{6.6.9}$$

6.7 Appendix B: Derivation of the term σ_f^2

Note that all of the following derivation is based on the condition $L_{opt} \leq L(\infty) \leq L_{opt}+\Delta$, and the terms G and H are equal to 0 at steady-state.

The variance of the term $\alpha+\gamma(A-B+C-D+E-F+G-H)$ is

$$\sigma_f^2 = E\{(\alpha+\gamma(A-B+C-D+E-F))-E\{\alpha+\gamma(A-B+C-D+E-F)\}\}^2 \tag{6.7.1}$$

From (6.2.9) we have

$$E\{\alpha+\gamma(A-B+C-D+E-F)\}=0 \tag{6.7.2}$$

Substituting equations (6.2.9) and (6.7.2) into (6.7.1), and using assumptions *A.2*, *A.3*, *A.4* and the mathematical formulation of terms *A*, *B*, *C*, *D*, *E* and *F*, it is straightforward to obtain that

$$\sigma_f^2 = \gamma^2(\sigma_A^2+\sigma_B^2+\sigma_C^2+\sigma_D^2+\sigma_E^2+\sigma_F^2-2E\{EF\})-\alpha^2 \tag{6.7.3}$$

With assumptions *A.2* and *A.4* and using equation (6.6.9) in the mathematical formulation of term A, its variance can then be obtained

$$\sigma_A^2 = \frac{4\Delta\mu'\sigma_v^4}{(2-\mu')L_{opt}} \tag{6.7.4}$$

Similarly, with assumptions *A.2* and *A.3*, and using (6.2.11) in the mathematical formulation of term B, its variance can be derived as

$$\sigma_B^2 \approx 4\sigma_v^2(L_{opt}+\Delta-L(\infty))\sigma_c^2\sigma_x^2 \tag{6.7.5}$$

Substituting (6.2.16) into (6.7.5) yields

$$\sigma_B^2 \approx 4\left(\frac{\alpha\sigma_v^2}{\gamma}+\frac{\Delta\mu'\sigma_v^4}{(2-\mu')L_{opt}}\right) \tag{6.7.6}$$

Using assumptions *A.2* and *A.4* in the mathematical formulation

of term C, its variance can be obtained

$$\sigma_C^2 \approx 4\Delta(L(\infty) - \Delta)\sigma_g^4\sigma_x^4 \tag{6.7.7}$$

With *A.4* and using equation (6.6.9) the following equation can be obtained

$$\sigma_g^2 \approx \frac{\mu'\sigma_v^2}{(2-\mu')L_{opt}\sigma_x^2} \tag{6.7.8}$$

Substituting (6.7.8) into (6.7.7), and with the approximation $L_{opt} \approx L(\infty) - \Delta$ results in

$$\sigma_C^2 \approx \frac{4\Delta\mu'^2\sigma_v^4}{(2-\mu')^2 L_{opt}} \tag{6.7.9}$$

Similarly, with assumptions *A.2*, *A.3* and *A.4*, and using equations (6.2.11), (6.2.16) and (6.7.8) in the mathematical formulation of term D, and with the approximation $L_{opt} \approx L(\infty) - \Delta$, the variance of term D can be obtained

$$\sigma_D^2 \approx \frac{4\mu'\sigma_v^2}{(2-\mu')}\left(\frac{\alpha}{\gamma} + \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}}\right) \tag{6.7.10}$$

Since the term $\mathbf{x}''^T(n)\mathbf{g}''(n)$ can be approximately deemed as a sum of Δ i.i.d random variables, from the central limit theory the probability density function (PDF) of this term will be very close to a Gaussian distribution with a zero mean, thus term E can be approximately deemed as chi-squared distributed with one degree of freedom. With assumptions *A.2* and *A.4*, and using equation (6.6.9) the mean value of term

E can be formulated as

$$m_E \approx \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}} \tag{6.7.11}$$

Thus the variance of term E is

$$\sigma_E^2 = 2m_E \approx 2\frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}} \tag{6.7.12}$$

Similarly, the term $\mathbf{x}''^T(n)\mathbf{c}''$ can be approximately deemed as Gaussian distributed, thus term F can be approximately deemed as chi-squared distributed with one degree of freedom. With assumptions *A.3* and *A.4*, and using equations (6.2.11) and (6.2.16) the mean value of term F can be formulated as

$$m_F \approx \frac{\alpha}{\gamma} + \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}} \tag{6.7.13}$$

Thus the variance for term F is

$$\sigma_F^2 = 2m_F = 2\left(\frac{\alpha}{\gamma} + \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}}\right) \tag{6.7.14}$$

With assumptions *A.3* and *A.4* the term $E\{EF\}$ can be expanded as

$$\begin{aligned} E\{EF\} &= E\{(\mathbf{x}''^T(n)\mathbf{g}''(n))^2(\mathbf{x}''^T(n)\mathbf{c}'')^2\} \\ &= E\{\mathbf{x}''^T(n)\mathbf{g}''(n)\mathbf{g}''^T(n)\mathbf{x}''(n)\mathbf{x}''^T(n)\mathbf{c}''\mathbf{c}''^T\mathbf{x}(n)\} \\ &\approx \sigma_g^2\sigma_c^2 E\{\mathbf{x}''^T(n)\mathbf{x}''(n)\mathbf{x}''^T(n)\mathbf{x}''(n)\} \end{aligned} \tag{6.7.15}$$

by approximating the instantaneous term $\mathbf{c}''\mathbf{c}''^T$ with its statistical averaging value.

The input signal is an i.i.d Gaussian sequence, thus $E\{x^4\} = 3\sigma_x^2$. From equation (6.7.15) it is straightforward to obtain

$$E\{EF\} = \sigma_g^2\sigma_c^2(\Delta(\Delta-1)\sigma_x^4 + 3\Delta\sigma_x^2) \tag{6.7.16}$$

Substituting (6.7.8) into (6.7.16) yields

$$E\{EF\} = \frac{\Delta\mu'\sigma_v^2\sigma_c^2((\Delta-1)\sigma_x^2+3)}{(2-\mu')L_{opt}} \tag{6.7.17}$$

Substituting equations (6.7.4), (6.7.6), (6.7.9), (6.7.10), (6.7.12), (6.7.14) and (6.7.17) into (6.7.3) yields

$$\sigma_f^2 = \gamma^2(K_2\frac{\alpha}{\gamma} + K_3) - \alpha^2 \tag{6.7.18}$$

where

$$K_1 = \frac{\Delta\mu'\sigma_v^2}{(2-\mu')L_{opt}} \tag{6.7.19}$$

and

$$K_2 = 2 + 4\sigma_v^2 + \frac{4L_{opt}K_1}{\Delta} \tag{6.7.20}$$

and

$$K_3 = 2K_1K_2 - 2K_1\sigma_c^2((\Delta-1)\sigma_x^2+3) \tag{6.7.21}$$

Chapter 7

VARIABLE TAP-LENGTH NATURAL GRADIENT ALGORITHM

As has been introduced in Chapter 3, in the convolutive blind source separation (BSS) problem, if there is only one signal and one received convolutive mixture signal, the BSS problem then becomes the blind deconvolution, or blind equalization, problem. Blind deconvolution is an important task for many applications in the communications and signal processing areas. The NG algorithm is a computationally efficient blind deconvolution algorithm [31]. In all the previous formulations of the NG algorithm, or modified NG algorithms, the tap-length of the deconvolution filter is assumed fixed. However, in many applications it is difficult to choose a good value for the deconvolution filter tap-length. If the tap-length is too long, the computational complexity will increase, whilst if the tap-length is too short, it will be inadequate for the deconvolution task. An excessive adaptive filter length can also be problematic due to the gradient noise, which is proportional to the tap-length. A variable tap-length NG algorithm is therefore needed to establish a good choice for the tap-length. In this chapter, the con-

cept of variable tap-length is for the first time introduced into the blind adaptive schemes, particularly the NG algorithm in single channel blind deconvolution or equalization applications.

This chapter is organized as follows: the NG algorithm for single channel blind deconvolution is introduced in Section 7.1. The proposed variable tap-length NG algorithm is described in Section 7.2. Some discussions for the proposed algorithm are given in Section 7.3. Section 7.4 provides the conclusion.

7.1 Introduction of the NG algorithm for blind deconvolution

Blind deconvolution is aimed at recovering a desired discrete-time signal $s(n)$ from its filtered and noisy version formulated as

$$x(n) = \sum_{i=1}^{P} h_i s(n-i) + v(n) \tag{7.1.1}$$

where h_i is the ith element of the unknown P-tap filter vector $\mathbf{h}$, $v(n)$ is the zero-mean additive noise, and n denotes the discrete time index. The NG blind deconvolution algorithm updates the adaptive filter coefficient vector $\mathbf{w}(n)$ so that on convergence the output of the adaptive filter $y(n)$ is ideally a delayed and possibly scaled version of the original signal $s(n)$. The criterion for the NG BSS algorithm has been introduced in Chapter 3. As a special case, for the blind deconvolution NG algorithm, in which the number of the source signals and mixture signals is unity, the update of the deconvolution filter vector $\mathbf{w}(n)$ can be formulated as

$$y(n) = \sum_{i=0}^{Q} w_i(n) x(n-i) \tag{7.1.2}$$

$$u(n) = \sum_{i=0}^{Q} w_{Q-i}(n) y(n-i) \tag{7.1.3}$$

$$w_i(n+1) = w_i(n) + \mu[(w_i(n) - f(y(n-Q))u(n-i)], \quad i = 0, ..., Q \tag{7.1.4}$$

where $w_i(n)$ is the ith element of the deconvolution filter vector $\mathbf{w}(n)$ with a tap-length of Q, μ is a positive step size, and $f(\cdot)$ is the nonlinear function, as has been discussed in Chapter 3. The tap-length Q used in the NG algorithm provides a trade-off between the computational complexity and the steady-state performance, and a variable tap-length NG algorithm is needed to provide a good choice for the tap-length.

7.2 A variable tap-length NG algorithm

Similar to that of the fractional tap-length (FT) algorithm introduced in Chapter 6, the tap-length of the NG algorithm is made variable so that at steady-state the tap-length converges to a value which can provide a good trade-off between the computational complexity and steady-state performance. The update of the proposed algorithm can be formulated as follows:

$$y(n) = \sum_{i=0}^{L(n)} w_{i,L(n)}(n) x(n-i) \tag{7.2.1}$$

$$u(n) = \sum_{i=0}^{L(n)} w_{L(n)-i,L(n)}(n) y(n-i) \tag{7.2.2}$$

$$w_i(n+1) = w_{i,L(n)}(n) + \mu[(w_{i,L(n)}(n) - f(y(n-L(n)))u(n-i)], \quad i = 0, ..., L(n) \tag{7.2.3}$$

where $w_{i,L(n)}(n)$ is the ith coefficient of the adaptive filter coefficient vector $\mathbf{w}(n)$ with a tap-length of $L(n)$. It is clear to see that the only difference between the proposed algorithm and the NG algorithm is

that the tap-length of the deconvolution filter vector of the proposed algorithm is a variable value rather than a fixed value.

The tap-length in the proposed algorithm should be updated to maximize the measure of the independence of the separated signals. As is well known, the normalized kurtosis is a measure of nonGaussianity, and also an approximate measure of the independence of signals [17]. The normalized kurtosis of the real output signal can be formulated as

$$K = \frac{E\{[y(n)]^4\}}{E\{[y(n)]^2\}^2} - 3 \tag{7.2.4}$$

where $E\{\cdot\}$ denotes the statistical expectation operator. In the NG algorithm for recovering nonGaussian signals, the absolute value of K should be maximized. To update the tap-length $L(n)$, the cost function to establish the optimal tap-length L_{opt} is defined as the minimum value L such that

$$|K| - |K'| \leq \varepsilon \tag{7.2.5}$$

where ε is a small positive value, $|\cdot|$ is the absolute value operation, and K' is the normalized kurtosis of the output signal generated by the first $L(n) - \Delta$ coefficients of the adaptive filter vector

$$y'(n) = \sum_{i=0}^{L(n)-\Delta} w_{i,L(n)}(n)x(n-i) \tag{7.2.6}$$

where Δ is a positive integer.

The motivation of this cost function is similar to that in [12]: the optimal tap-length of the adaptive filter is defined as the minimum value of the tap-length which ensures the difference between K and K' satisfies the inequality in (7.2.6). To ensure that the update of the tap-

length is continuous, similar to the approach in [12], in which a variable tap-length LMS algorithm is proposed, the concept of fractional tap-length l_f is also used in the proposed algorithm. The update of this fractional tap-length can be described as

$$l_f(n+1) = l_f(n) - \alpha + \gamma\Delta_k \tag{7.2.7}$$

where α is a leakage parameter to avoid over estimation of the tap-length, γ is the step size parameter for the fractional tap-length update, and Δ_k is defined as

$$\Delta_k = |K| - |K'| \tag{7.2.8}$$

In practice, both the statistical values K and K' are replaced by time average instantaneous values K_{est} and K'_{est}. To obtain the time average values and reduce the computational complexity, both the fractional tap-length $l_f(n)$ and the tap-length $L(n)$ will only be updated every W iterations, and the estimates K and K' are obtained by averaging W instantaneous values

$$K_{est} = \frac{\frac{1}{W}\sum_{p=i}^{i-W+1}[y(p)]^4}{(\frac{1}{W}\sum_{p=i}^{i-W+1}[y(p)]^2)^2}, \quad if \;\; mod(n,W) = 0 \tag{7.2.9}$$

where $mod(n,W) = 0$ denotes that n is a multiple of W. The estimate K'_{est} is obtained similarly to K_{est} by replacing $y(p)$ with $y'(p)$ in (7.2.9). The statistical accuracy of this estimator has been found in later simulation to be sufficient for practical application. The practical update of the fractional tap length is then formulated as

$$l_f(n+1) = l_f(n) - \alpha + \gamma(|K_{est}| - K'_{est}|) \quad if \;\; mod(n,W) = 0 \tag{7.2.10}$$

The tap-length $L(n+1)$, which will be used in the update of the adaptive filter coefficients in the next W iterations, is obtained from the fractional tap-length $l_f(n+1)$ as

$$\begin{aligned} &if\ |l_f(n+1) - L(n)| > M\ and\ mod(n, W) = 0 \\ &\quad L(n+1) = \max[floor(l_f(n+1)), L_{min}] \\ &\quad else \\ &\quad L(n+1) = L(n) \end{aligned} \tag{7.2.11}$$

where $floor$ is the operation which rounds down the embraced value to the nearest integer, M is a small positive integer and L_{min} is a positive integer. Note that the tap-length is constrained to be not less than L_{min}, where $L_{min} > \Delta$, since $L(n) - \Delta$ is used as a tap-length in (7.2.6), and it should not be less than unity.

Next a simulation will be performed to show the performance of the proposed algorithm.

7.3 Simulation

One example simulation framework is used to compare the performance of the proposed variable tap-length NG algorithm with the original NG algorithm. In all the simulations the original signal $s(n)$ is a sub-Gaussian pseudo-random sequence which is uniformly-distributed in the range $[-0.5, 0.5]$. The channel which needs to be equalized is modelled by a finite impulse response (FIR) filter $\mathbf{h} = [1.0, 0.8, -0.75, 0.5, -0.4, 0.3, 0.2, 0.15, -0.07, 0.1]$. The noise signal is a pseudo-random zero-mean Gaussian sequence, and scaled to make the signal-to-noise ratio (SNR) 20dB. The performance of the proposed algorithm and the NG

algorithm with different tap-lengths $L = 10, 15, 20$ and 25 is examined by measuring the inter-symbol interference (ISI), defined as

$$ISI(n) = \frac{\sum_{i=1}^{N-1} g_i^2(n)}{\max_{0 \le j \le N-1} g_i^2(n)} - 1 \qquad (7.3.1)$$

where $g_i^2(n)$ is the squared ith element of $\mathbf{g}(n)$, which is the convolution of the channel vector $\mathbf{h}$ and the adaptive filter vector $\mathbf{w}(n)$, and N is its tap-length. For both the NG algorithm and the proposed variable tap-length NG algorithm the nonlinear function is chosen as $f(y) = y^3$, $w_i(0) = \delta_{i-4}$ (i.e. the fourth coefficient of the adaptive filter is initialized to unity, and the others are zeros), and $\mu = 0.001$. For the proposed algorithm, the parameters for updating the fractional tap-length are set as $\gamma = 1$, $\alpha = 0.001$, $M = 1$, $\Delta = 8$, $W = 100$ and $L_{min} = 10$. The initial value of the tap-length is set to 10. One hundred Monte Carlo trails are run and the results are obtained by averaging all the cases. The evolution curves of the ISI for the proposed algorithm and the original NG algorithm with different tap-lengths are shown in Fig. 7.1(a). The evolution curve of the fractional tap-length for the proposed algorithm is shown in Fig. 7.1(b).

It is apparent to see in Fig. 7.1(a) that with a tap-length $L = 10$ the performance of the NG algorithm is poor, and the ISI level is high, which indicates that the tap-length value 10 is inadequate for the deconvolution. The performance of the NG algorithm with tap-length value 15 is generally good enough for most applications (ISI level is less than -20dB). The performances of the NG algorithm with tap-lengths 20 and 25 are very similar, which indicates that with the increase of the tap-length, the improvement of the performance is very limited,

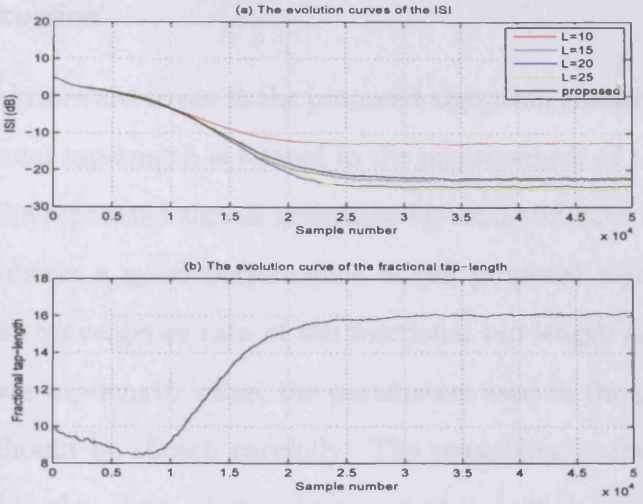

Figure 7.1. Performance of the proposed algorithm and the NG algorithm with different tap-length values.

and the computational complexity will increase. Thus values around 15 will be good choices for the tap-length of the NG algorithm, which provide good compromise between the performance and the computational complexity.

From Fig. 7.1(b) it is clear to see that the fractional tap-length of the proposed algorithm converges to values close to 16, and this steady-state tap-length provides good performance with the proposed algorithm, which can be seen in Fig. 7.1(a), and a lower computational complexity as compared with tap-length values 20 and 25, since equation (7.2.7) is only evaluated twice, and equations (7.2.8) and (7.2.9) are evaluated once for each W measurement samples (equation (7.2.7) is the most significant term requiring approximately two additions and multiplications per sample). Note that the initial convergence of the fractional tap-length in Fig. 7.1(b) is because the coefficients of the adaptive filter are still to converge.

7.4 Discussion

It is straightforward to see in the proposed algorithm that the update of the fractional tap-length is related to the measurement of the independence of the separated signals generated by using different tap-lengths. Thus to obtain a good performance of the proposed algorithm, i.e., both a fast convergence rate of the fractional tap-length and a proper steady-state tap-length value, the parameters used in the proposed algorithm should be chosen carefully. The sensitivity in their selection falls outside the scope of this demonstrative example, but it is clear that appropriate settings can be found.

7.5 Conclusion

A new variable tap-length NG algorithm for single channel blind deconvolution has been proposed in this chapter. This is the first example of a variable tap-length sequential blind adaptive algorithm. As shown by the simulations, the tap-length of the proposed algorithm converges to values which provide good compromise between the steady-state ISI and the computational complexity. This algorithm can be potentially used in many applications.

Chapter 8

CONCLUSION

In this final chapter of the thesis, the work and results presented in the previous chapters are summarized. Overall conclusions of this study are made and further work is suggested.

8.1 Summary of the thesis

This thesis concerns estimation of the reverberation time (RT) in high noise occupied rooms. Room RT is a very important acoustic parameter for characterizing the quality of an auditory space. The RTs of occupied rooms are particularly important since they are more close to reality by considering the existence of the audience. Some traditional RT estimation methods which normally utilize high sound pressure noises as excitation signals are not suitable for occupied rooms, since for the audience, exposure to loud noise for a long period can be disturbing. The methods utilizing passively received signals, especially speech signals, are more attractive for occupied rooms, since good controlled excitation signals are not necessary. However, the accuracy of these methods may be influenced by the noise, which is generated by the audience. Thus new approaches are needed to improve the RT estimation accuracy in a high noise environment.

In this thesis, the maximum likelihood estimation (MLE) based RT

estimation method is firstly described. A background introduction to the adaptive techniques, i.e., the blind source separation (BSS) scheme and the least mean square (LMS) algorithm, is also provided. By utilizing the BSS algorithm, the LMS algorithm and the MLE based RT estimation method, a new framework for high noise environment RT estimation is proposed. The motivation of the proposed approach is to reduce the noise level from the passively received speech signal before the RT estimation. Since both the excitation speech signal and the noise signal are unknown, the proposed noise reducing preprocessing can be deemed as a blind process, in which the BSS technique can be utilized. Ideally, the BSS technique can extract estimations of the original excitation signal and noise signal, from which estimations of room impulse responses can be obtained, and consequently the room RT can be calculated. However, in practice, the convolutive BSS algorithm can only extract an estimation of an unknown filtered version of source signals. To remove the noise component from the passively received speech signal, an extra ANC stage is needed by using the estimation of the noise signal coming from the BSS as a reference signal. The output of the ANC is then an estimation of the noise free reverberant speech signal, from which a more accurate RT estimation can potentially be extracted. As shown by the simulation results, the noise reducing preprocessing works well in a simulated high noise room, and accuracy of the RT estimate is improved as compared with the original MLE based RT estimation method.

Further research results on adaptive techniques are also provided in this thesis. At first, to speed up the convergence rate of the LMS algorithm, the concept of gradient based variable step size LMS (VSSLMS)

algorithms is introduced. Two new gradient based VSSLMS algorithms are proposed. Simulations are performed which show that the proposed algorithms are robust to high level, a signal-to-noise ratio (SNR) of approximately 0dB, statistical stationary or nonstationary noise. Although both algorithms can not be used directly in the ANC stage of the proposed RT estimation framework due to the statistical nonstationary of both the input and noise signals, these research results provide a deeper understanding of the VSSLMS algorithms, and may be potentially used in other applications.

It is clear to see that in the ANC stage, the optimal tap-length for the adaptive filter is unknown. To search for a good choice of the steady-state adaptive filter tap-length, variable tap-length LMS (VTLMS) algorithms are needed. New research results have been obtained for VTLMS algorithms, i.e., a steady-state performance analysis of the FT algorithm, which is a robust VTLMS algorithm; improvement of the convergence performance of the FT algorithm in a high noise condition by utilizing a convex combination approach; and a new practical variable tap-length LMS algorithm for applications in which the optimal filter has an exponential decay impulse response. All the analysis and proposed algorithms are confirmed and supported by simulations. Again, although these research results can not be used directly in the proposed RT estimation framework due to the statistical nonstationary of both the input and noise signals, they are potentially very useful in many applications where the optimal tap-length of the LMS algorithm is unknown.

The idea of variable tap-length is also introduced for the first time into the BSS research area, in particular for a key sequential BSS algo-

rithm, the NG algorithm. A variable tap-length NG algorithm is proposed to search for a good choice of the adaptive filter tap-length. As shown by the simulation results, the proposed algorithm can converge to a tap-length which provides a good trade-off between the steady-state performance and computational complexity. Due to the similarity of the optimal tap-length model for the LMS algorithm and the BSS algorithms, more research is required for variable tap-length BSS algorithms. The idea of variable tap-length can be potentially extended to multi-channel BSS problems.

8.2 Overall conclusions and future work

It has been shown that the proposed RT estimation framework performs well under a simulated high noise environment. The accuracy of the RT estimations is improved due to the noise reducing preprocessing. The research results on VSSLMS algorithms are very useful for a high noise environment. The research results on VTLMS algorithms and the variable tap-length NG algorithm provide new blind signal processing techniques for applications where the optimal tap-length of the adaptive filter is unknown. To make the proposed RT estimation approach more practical in reality, future research is suggested:

1. The key step of the proposed method is clearly the BSS stage. Normally, the outputs of BSS are not exactly filtered versions of the source signals, especially in a high RT environment, due to certain fundamental limitations, such as data length restrictions and modelling uncertainties. In practice, the outputs of BSS contain components from different source signals. It has been shown in the thesis that when the RT is less than 0.3s, the performance of BSS can be good for the RT

estimation in a simulated high noise room. However, the room RT in many applications is larger than 0.3s, and therefore the performance of convolutive BSS algorithm is still poor. The improvement of the performance of convolutive BSS algorithms for long RT rooms is then the most valuable work for the proposed framework. Note that this work is also important to the BSS problem itself.

2. The existing VSSLMS algorithms and VTLMS algorithms, including the research results presented in this thesis are not suitable for applications where both input and noise signals are statistical nonstationary. New VSSLMS algorithms and VTLMS algorithms are needed to obtain robustness to such signals.

3. The concept of variable tap-length can be potentially utilized in other BSS algorithms, including muti-channel convolutive BSS algorithms. More theoretical research is needed to investigate the relationship between the performance of the BSS algorithms and their adaptive filter vector/matrix tap-length/order.

BIBLIOGRAPHY

[1] Y. Gu, K. Tang, H. Cui, and W. Du, "Convergence analysis of a deficient-length LMS filter and optimal-length sequence to model exponential decay impulse response," *IEEE Signal Processing Lett.*, vol. 10, pp. 4–7, Jan. 2003.

[2] W. C. Sabine, *Collected papers on acoustics.* Harvard U.P., 1922.

[3] F. F. Li, *Extracting room acoustic parameters from received speech signals using artificial neural networks.* PhD thesis, Salford University, 2002.

[4] H. Kuttruff, *Room Acoustics 4th ed.* London: Spon Press, 2000.

[5] I. 3382, "Acoustics-measurement of the reverberation time of rooms with reference to other acoustical parameters," *International Organization for Standardization*, 1997.

[6] M. R. Schroeder, "New method for measuring reverberation time," *J. Acoust. Soc. Amer.*, vol. 37, pp. 409–412, 1965.

[7] T. J. Cox, F. Li and P. Darlington, "Extracting room reverberation time from speech using artificial neural networks," *J. Audio. Eng. Soc*, vol. 49, pp. 219–230, 2001.

[8] R. Ratnam, D. L. Jones, B. C. Wheeler, W. D. O'Brien Jr., C. R. Lansing and A. S. Feng, "Blind estimation of reverberation time," *J. Acoust. Soc. Amer.*, vol. 114, pp. 2877–2892, Nov. 2003.

[9] L. Parra and C. Spence, "Convolutive blind source separation of non-stationary sources," *IEEE Trans. Speech and Audio Processing*, vol. 8, pp. 320–327, May. 2000.

[10] W. Wang, S. Sanei and J. A. Chambers, "Penalty function-based joint diagonalization approach for convolutive blind separation of non-stationary sources," *IEEE Trans. Signal Processing*, vol. 53, pp. 1654–1669, May 2005.

[11] O. Macchi, *Adaptive Processing: The Least Mean Squares Approach with Applications in Transmission.* New York, Wiley, 1995.

[12] Y. Gong and C. F. N. Cowan, "An LMS style variable tap-length algorithm for structure adaptation," *IEEE Trans. Signal Processing*, vol. 53, pp. 2400–2407, July 2005.

[13] R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification 2nd ed.* John Wiley & Sons, 2001.

[14] R. Ratnam, D. L. Jones and W. D. O'Brien Jr., "Fast algorithms for blind estimation of reverberation time," *IEEE Signal Processing Lett.*, vol. 11, pp. 537–540, June 2004.

[15] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications.* New York, Wiley, 1998.

[16] A. H. Sayed, *Fundamentals of Adaptive Filtering.* New York, Wiley, NJ, 2003.

[17] A. Hyvarinen, J. Karhunen and E. Oja, *Independent component analysis.* Wiley, 2001.

[18] S. Haykin, *Unsupervised adaptive filtering.* Wiley, 2000.

[19] C. Jutten, and J. Hérault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, pp. 1–10, 1991.

[20] P. Comon, "Independent component analysis: A new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.

[21] A. J. Bell and T.J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.

[22] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Advances in Neural Information Processing Systems 8*, MIT Press, 1996.

[23] J. F. Cardoso, "Infomax and maximum likelihood for blind source separation," *IEEE Signal Processing Lett.*, vol. 4, pp. 112–114, Apr. 1997.

[24] L. Molgedey and H. Schuster, "Separation of a mixture of independent signals using time delayed correlations," *Phys. Rev. Lett.*, vol. 72, pp. 3634–3637, Oct. 1994.

[25] L. Tong, R. W. Liu, V. C. Soon and Y. F. Huang, "Inderterminacy and identifiability of blind identification," *IEEE Trans. Circuits and Systems*, vol. 38, pp. 499–509, 1991.

[26] A. Belouchrani, K. Abed Meraim, J. F. Cardoso, and E. Moulines, "A blind source separation technique based on second order statistics," *IEEE Trans. Signal Processing*, vol. 45, pp. 434–444, 1997.

[27] K. Matsuoka, M. Ohya, and M. Kawamoto, "A neural net for blind separation of nonstationary signals," *Neural Networks*, vol. 8, pp. 411–419, 1995.

[28] J. F. Cardoso, "Blind signal separation:statistical principles," *Proc. of IEEE*, vol. 86, pp. 2009–2025, Oct. 1998.

[29] S. Amari and A. Cichocki, "Adaptive blind signal processing - neural network approaches," *Proc. of IEEE*, vol. 86, pp. 2026–2048, Oct. 1998.

[30] T. W. Lee, *Independent component analysis-theory and application.* Kluwer academic publishers, 1998.

[31] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, "Multichannel blind deconvolution and equalization using the natural gradient," in *Proc. 1st IEEE Workshop on Signal Processing Advances in Wireless Communications*, Apr. 1997.

[32] T. W. Lee, A. J. Bell, and R. Lambert, "Blind separation of delayed and convolved sources," in *Advances in Neural Information Processing Systems 9*, pp. 758–764, MIT Press, 1997.

[33] K. Torkolla, "Blind separation of convolved sources based on information maximization," in *IEEE workshop on Neural Networks and Signal Processing*, 1996.

[34] P. Smaragdis, "Information theoretic approaches to source separation," Master's thesis, MIT Media Lab, June 1997.

[35] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, 1998.

[36] S. Ikeda and N. Murata, "A method of ICA in time-frequency domain," in *International Conference on Independent Component Analysis and Signal Separation*, pp. 365–371, Jan 1999.

[37] N. Mitianoudis and M. Davies, "Audio source separation of convolutive mixtures," *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 489–497, Sep. 2003.

[38] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech," tech. rep., RIKEN Brain Science Institute, 1998.

[39] S. Araki, R. Mukai, S. Makino, T. Nishikawa and H. Saruwatari, "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Trans. Speech and Audio Processing*, vol. 11, pp. 109–116, Mar. 2003.

[40] L. Parra and C. V. Alvino, "Geometric source separation: merging convolutive source separation with geometric beamforming," *IEEE Trans. Speech and Audio Processing*, vol. 10, pp. 352–362, Sep. 2002.

[41] L. Parra and C. V. Alvino, "Geometric source separation: merging convolutive source separation with geometric beamforming," in *Proc. NNSP2001*, pp. 273–282, Sep. 2001.

[42] B. V. Veen and K. Bckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, pp. 4–24, Apr. 1998.

[43] S. Araki, S. Makino, R. Mukai, Y. Hinamoto, T. Nishikawa and H. Saruwatari, "Equivalence between frequency domain blind source separation and frequency domain adaptive beamforming," in *Proc. ICASSP2002*, pp. 1785–1788, May. 2002.

[44] S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura, "Evaluation of blind signal separation method using directivity pattern," in *Proc. ICASSP2000*, pp. 3140–3143, June 2000.

[45] M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," in *Proc. ICASSP2002*, pp. 881–884, May 2002.

[46] M. Knaak, S. Araki, and S. Makino, "Geometrically constrained independent component analysis," *IEEE Trans. Speech and Audio Processing*, vol. 15, pp. 715–726, Feb. 2007.

[47] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech and Audio Processing*, vol. 12, pp. 530–538, Sep. 2001.

[48] J. E. Greenberg, "Modified LMS algorithm for speech processing with an adaptive noise canceller," *IEEE Trans. Signal Processing*, vol. 6, pp. 338–351, July 1998.

[49] D. L. Duttweile, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech and Audio Processing*, vol. 8, pp. 508–518, Sep. 2000.

[50] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, pp. 943–950, Apr. 1979.

[51] S. Karni and G. Zeng, "A new convergence factor for adaptive filters," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 1011–1012, July 1989.

[52] R. H. Kwong and E. W. Johnson, "A variable step size LMS algorithm," *IEEE Trans. Signal Processing*, vol. 40, pp. 1633–1642, July 1992.

[53] V. J. Mathews and Z. Xie, "A stochastic gradient adaptive filter with gradient adaptive step size," *IEEE Trans. Signal Processing*, vol. 41, pp. 2075–2087, June 1993.

[54] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: analysis and simulations," *IEEE Trans. Signal Processing*, vol. 45, pp. 631–639, Mar. 1997.

[55] D. I. Pazaitis and A. G. Constantinides, "A novel kurtosis driven variable step-size adaptive algorithm," *IEEE Trans. Signal Processing*, vol. 47, pp. 864–872, Mar. 1999.

[56] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic echo cancellation filtersan overview," *Signal Processing*, vol. 80, pp. 1697–1719, Sep. 2000.

[57] W. Ang and B. Farhang-Boroujeny, "A new class of gradient adaptive step-size LMS algorithms," *IEEE Trans. Signal Processing*, vol. 49, pp. 805–810, Apr. 2001.

[58] H.-C. Shin, A. H. Sayed, and W.-J. Song, "Variable step-size NLMS and affine projection algorithms," *IEEE Signal Processing Lett.*, vol. 11, pp. 132–135, Feb. 2004.

[59] K. Mayyas, "Performance analysis of the deficient length LMS adaptive algorithm," *IEEE Trans. Signal Processing*, vol. 53, pp. 2727–2734, Aug. 2005.

[60] F. Riero-Palou, J. M. Noras, and D. G. M. Cruickshank, "Linear equalisers with dynamic and automatic length selection," *Electronics Lett.*, vol. 37, pp. 1553–1554, Dec. 2001.

[61] Y. Gu, K. Tang, H. Cui, and W. Du, "LMS algorithm with gradient descent filter length," *IEEE Signal Processing Lett.*, vol. 11, pp. 305–307, Mar. 2004.

[62] Y. Gong and C. F. N. Cowan, "A novel variable tap-length algorithm for linear adaptive filters," in *Proc. ICASSP2004*, Jan. 2004.

[63] Y. Gong and C. F. N. Cowan, "Structure adaptation of linear MMSE adaptive filters," *Proc. Inst. Elect. Eng.-Vision, Image, Signal Processing*, vol. 151, pp. 271–277, Aug. 2004.

[64] M. Tarrab and A. Feuer, "Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data," *IEEE Trans. Information Theory*, vol. 34, pp. 680–691, July 1988.

[65] J. Arenas-García, A. R. Figueiras-Vidal, and A. H. Sayed, "Steady-state performance of convex combinations of adaptive filters," in *Proc. ICASSP2005*, Mar. 2005.

[66] J. Arenas-García, V. Gómez-Verdejo, and A. R. Figueiras-Vidal, "New algorithms for improved adaptive convex combination of LMS transversal filters," *IEEE Trans. Instrumentation and Measurement*, vol. 54, pp. 2239–2249, Dec. 2005.