People's Democratic Republic of Algeria

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

ABOU-BEKR BELKAID UNIVERSITY – TLEMCEN

FACULTY OF SCIENCES – DEPARTMENT OF ELECTRICAL ENGINEERING

THESIS

Presented in: TLEMCEN

To obtain the diploma of:

DOCTORATE

In Electronics

Option: Instrumentation

by:

Mrs. Aicha Benyahia

On the theme

Advanced Two-Sensor Adaptive Algorithms for Noise Reduction in Dispersive and Sparse Acoustic Environments

---

Publicly defended on 01 April 2026 in Tlemcen before the jury composed of:

Mrs. Ahlam GUEN	Professor University of Tlemcen	Chair
Mr. Rédha Bendoumia	Senior Lecturer (A) University of Blida 1	Thesis Director
Mrs. Nadéra Kaddouri	Senior Lecturer (B)University of Tlemcen	Co-Supervisor
Mr. Salim Kerai	ProfessorUniversity of Tlemcen	Examiner
Mr. Rachid Merzougui	Professor University of Tlemcen	Examiner
Mr. Djalel Ziani Kerarti	Senior Lecturer (A) ENSTTIC, Oran	Examiner

Laboratory Research Unit of Materials and Renewable Energy (URMER), University of Tlemcen, BP 119, 13000

Tlemcen – Algérie

Dédicaces

To the **loving memory of my dear mother**,
whose kindness, love, and sacrifices continue to guide my steps.
May Allah grant her His infinite mercy and bless her with the highest place in Paradise.

To my **beloved father**,
whom I pray Allah grants a long life, good health, and peace.
Thank you for your strength, wisdom, and unwavering support.

To my **dear husbandBoumediene**,
for your patience, encouragement, and constant presence throughout this journey.

To my **daughters, Sarah and Hanane**,
the source of my joy, pride, and motivation.

To my **son, Abderrahmane**,
the heartbeat that inspires me every day.

I dedicate this thesis to those who give true meaning to my life.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to **God Almighty**, who has guided me on the right path, granted me health, and enabled me to accomplish this humble

work under the best possible conditions.

I would like to extend my sincere appreciation to my thesis supervisor, **Mr. Rhéda Bendoumia,** Senior Lecturer at Saad Dahleb University of Blida 1, for having accepted to supervise this work. His constant encouragement, his trust, his unfailing support, and his valuable guidance throughout this research have been essential to my academic and personal growth. I am deeply indebted to him for his time, his availability, and his patience in directing me toward the path of scientific research. Please find here the expression of my highest respect and gratitude.

I would like to equally express my sincere gratitude to my co-supervisor, **Mrs Nadéra Kaddouri**, for her valuable contribution, guidance, and support throughout this work. Her involvement and dedication have been greatly appreciated.

My profound thanks go to **Mrs Ahlam GUEN**, for the great honor she has done me by agreeing to chair the examination committee.

I am equally honored by the presence of **Mr. Salim Kerai**, **Mr. Djalel Ziani Kerarti**, and **Mr. Rachid Merzougui**, all from Abou Bakr Belkaid University of Tlemcen, who have kindly accepted to evaluate and review this work as examiners. I sincerely thank them for the attention, time, and interest they have devoted to my research.

My heartfelt gratitude goes to my family, whose unwavering trust, constant encouragement, and affection have been my greatest source of strength throughout this journey. Their support has helped me overcome the most difficult moments.

Finally, I would like to extend my warm thanks to all those who, in various ways and at different times, have offered me their help, advice, and encouragement during the completion of this work. Each of them, through their kindness and presence, has contributed to making this achievement possible.

Abstract

In this thesis, we address the problem of acoustic noise reduction and speech enhancement in modern telecommunication systems using two-sensors adaptive filtering and intelligent learning-based approaches. The research focuses on the development of advanced two-sensor adaptive algorithms capable of efficiently separating speech from noise in dispersive and sparse acoustic environments, where conventional methods often fail. In the first part of this project, we propose a novel Neural Network-based Variable Step-Size Feed-forward NLMS (NN-V-FNLMS) algorithm. This approach integrates a simple neural network to dynamically estimate the adaptation step-size, thus overcoming the inherent trade-off between fast convergence and low steady-state error found in conventional algorithms. A Voice Activity Detector (VAD) is also incorporated to control filter updates and improve computational efficiency. To further enhance robustness and adaptability, we propose a contribution that introduces an advanced Deep Learning Variable Step-Size Feed-forward NLMS (DL-VSS-FNLMS) algorithm. This model employs a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) layers to predict optimal step-size parameters based on rich acoustic features, including Mel-Frequency Cepstral Coefficients (MFCCs), GammatoneCepstral Coefficients (GTCCs), and spectral representations on the ERB, Bark, and Mel scales. This deep model dynamically adjusts the adaptation process in complex acoustic conditions, achieving superior performance in both dispersive and sparse environments. Experimental evaluations demonstrate that the proposed algorithms significantly improve convergence speed, stability, and speech quality, while ensuring better noise suppression compared to conventional LMS/NLMS and classical VSS algorithms. The results confirm that the integration of neural networks and deep learning into two-sensor adaptive filtering offers a powerful and flexible solution for real-world acoustic noise reduction and speech enhancement.

Keywords

Adaptive Filtering; Two-sensors Noise Reduction; Variable Step-Size Parameters; Recurrent Neural Networks; Deep Learning; Long Short-Term Memory (LSTM); Voice Activity Detection (VAD); MFCC/GTCC features; Dispersive and sparse environments.

الملخص

في هذه الرسالة، نتناول مشكلة تقليل الضوضاء الصوتية وتحسين جودة الكلام في أنظمة الاتصالات الحديثة باستخدام الترشيح التكيفي وأساليب التعلم الذكي. يركز البحث على تطوير خوارزميات تكيفية متقدمة ثنائية المستشعرات قادرة على فصل الكلام عن الضوضاء بكفاءة في البيئات الصوتية المشتتة والمكثفة، حيث تقدم الطرق التقليدية غالبًا أداء متواضع. في الجزء الأول من هذا المشروع، نقترح خوارزمية جديدة قائمة على الشبكات العصبية البسيطة، تعتمد على التغذية الأمامية متغيرة الحجم (NN-V-FNLMS).يدمج هذا النهج شبكة عصبية بسيطة لتقدير حجم خطوة التكيف ديناميكيًا، متغلبًا بذلك على المعادلة المتأصلة بين التقارب السريع وانخفاض خطأ الحالة المستقرة في الخوارزميات التقليدية. كما تم دمج كاشف نشاط الصوت (VAD) للتحكم في تحديثات المرشح وتحسين الكفاءة الحسابية. ولتعزيز المتانة والقدرة على التكيف بشكل أكبر، نقترح خوارزمية متقدمة قائمة على التعلم العميق، تعتمد على التغذية الأمامية متغيرة الحجم (DL-VSS-FNLMS).يستخدم هذا النموذج شبكة عصبية متكررة (RNN)مع طبقات ذاكرة طويلة و قصيرة المدى (LSTM) للتنبؤ بمعلمات حجم الخطوة الأمثل بناءً على خصائص صوتية غنية، بما في ذلك معاملات MFCCs وGTCCs، والتمثيلات الطيفية على مقاييس ERB وBark وMel. يُعدّل النموذج العميق عملية التكيف ديناميكيًا للتعامل مع الظروف الصوتية المعقدة، محققًا أداءً فائقًا في البيئات المشتتة والمتفرقة. تُظهر التقييمات التجريبية أن الخوارزميات المقترحة تُحسّن بشكل كبير سرعة التقارب، والاستقرار، وجودة الكلام، مع ضمان كبت أفضل للضوضاء مقارنةً بخوارزمياتالتقليدية وخوارزمياتالمتغيرةالكلاسيكية. تؤكد النتائج أن دمج الشبكات العصبية والتعلم العميق في التصفية التكيفية يوفر حلاً قويًا ومرنًا لتقليل الضوضاء الصوتية في العالم الحقيقي وتحسين الكلام.

الكلمات المفتاحية

الترشيح التكيفي؛ تقليل الضوضاء بمستشعرين؛ الخطوة المتغيرة؛ الشبكات العصبية المتكررة؛ التعلم العميق؛ الذاكرة طويلة وقصيرة المدى؛ نظام اكتشاف نشاط الصوت؛ ميزاتMFCC/GTCC ؛ البيئات المشتتة والمكثفة.

Résumé

Dans cette thèse, nous abordons le problème de la réduction du bruit acoustique et du rehaussement de la parole dans les systèmes de télécommunication modernes en utilisant des techniques de filtrage adaptatifbi-capteur et des approches d'apprentissage intelligent. Les travaux de recherche portent sur le développement des algorithmes adaptatifs avancés à deux capteurs, capables de séparer efficacement la parole du bruit dans des environnements acoustiques dispersifs et clairsemés, où les méthodes conventionnelles montrent leurs limites. Dans la première partie de ce travail, nous proposons un nouvel algorithme Forward NLMS à pas variable basé sur un réseau de neurone (NN-V-FNLMS). Cette approche intègre un réseau de neurone simple permettant d'estimer dynamiquement le pas d'adaptation, afin de surmonter le compromis inhérent entre la rapidité de convergence et la faible erreur en régime permanent des algorithmes classiques. Un détecteur d'activité vocale (VAD) est également intégré pour contrôler la mise à jour des filtres et améliorer l'efficacité computationnelle. Afin d'améliorer encore la robustesse et l'adaptabilité, nous proposons un algorithme Forward NLMS à pas variable basé sur l'apprentissage profond (DL-VSS-FNLMS). Ce modèle exploite un réseau de neurones récurrent (RNN) à mémoire à long et court terme (LSTM) pour prédire les paramètres optimaux du pas d'adaptation à partir de caractéristiques acoustiques riches, incluant les coefficients cepstraux en fréquence de Mel (MFCCs), les coefficients cepstrauxgammatone (GTCCs), ainsi que des représentations spectrales sur les échelles ERB, Bark et Mel. Le modèle profond ajuste dynamiquement le processus d'adaptation dans des conditions acoustiques complexes, offrant des performances supérieures aussi bien dans les environnements dispersifs que clairsemés. Les résultats démontrent que les algorithmes proposés améliorent la vitesse de convergence, la stabilité et la qualité de la parole, tout en assurant une meilleure suppression du bruit par rapport aux algorithmes LMS/NLMS conventionnels et aux approches classiques à pas variable. Les résultats confirment que l'intégration des réseaux de neurones et de l'apprentissage profond dans le filtrage adaptatif bi-capteurconstitue une solution puissante et flexible pour la réduction du bruit acoustique et l'amélioration de la parole dans des conditions réelles.

Mots-clés

Filtrage Adaptatif ; Réduction du Bruit Bi-capteurs ; Pas d'Adaptation Variables ; Réseaux Neuronaux Récurrents ; Apprentissage Profond ; Mémoire à Long et à Court Termes (LSTM) ; Détection d'Activité Vocale (VAD) ; MFCC/GTCC ; Environnements Dispersifs et Clairsemés.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Abbreviations

Adaptive Filtering & Signal Processing

DSP Digital Signal Processing
LMS Least Mean Squares algorithm
NLMS Normalized Least Mean Squares algorithm
FNLMS Feed-Forward NLMS
VSS Variable Step-Size
2-LMS Two-Channel LMS
2-NLMS Two-Channel NLMS
SAD Symmetric Adaptive Decorrelation
IR Impulse Response

Performance Metrics

MSE Mean Square Error
MAE MeanAbsoluteError
SM System Mismatch
SNR Signal-to-Noise Ratio
Seg-SNR Segmental Signal-to-Noise Ratio
$\mathbf{R^2}$ Correlation Coefficient

Nural Networks & Deep Learning

NN Neural Network
DL Deep Learning
DL-VSS Deep-Learning Variable Step-Size
NN-V-FNLMS Neural Network–based Variable Step-Size FNLMS
RNN Recurrent Neural Network
LSTM Long Short-Term Memory

Speech Processing

VAD Voice Activity Detector
MFCC Mel-FrequencyCepstral Coefficients
GTCC GammatoneCepstral Coefficients
FFT Fast Fourier Transform

List of Symbols

Signals& Variables

x(n) Input signal sample
d(n) Desired signal
y(n) Filter output
e(n) Error signal
b(n) Noise signal
$\hat{\mathbf{s}}\mathbf{(n)}$ Estimated clean speech
p(n) Primary microphone signal

Adaptive Filter Parameters

w(n) Adaptive filter coefficient vector
M Filterlength
μ, μ(n) Step-size / learning rate
μmax Maximum step-size
εSmall positive constant (stability)
$\|\mathbf{x(n)}\|^2$ Instantaneousenergy of input

Acoustic Mixing Model

$\mathbf{h_{12}}$, $\mathbf{h_{21}}$ Acoustic impulse responses
$\boldsymbol{\alpha_{12}}$, $\boldsymbol{\alpha_{21}}$ Mixing/scaling coefficients
IR Impulse Response type (dispersive / sparse)

Statistical Quantities

RxxAutocorrelation matrix of input
tr(Rxx) Trace of autocorrelation matrix
λmaxLargesteigenvalue of Rxx

Performance Measurement

SNR_in / SNR_out Input / Output SNR
MSE Mean Square Error
MAE MeanAbsoluteError
$\mathbf{R^2}$ Correlation Coefficient

General Introduction

Over the past few decades, digital signal processing has experienced remarkable growth due to its central role in various domains such as telecommunications, audio engineering, biomedical applications, and artificial intelligence. Among the many challenges faced in these fields, acoustic noise reduction remains a crucial issue, particularly in applications such as hands-free telephony, teleconferencing systems, voice assistants, and hearing aids. In real environments, speech signals captured by microphones are often corrupted by noise signals and by reverberation effects caused by the acoustic properties of the surrounding space. These distortions severely degrade speech quality and intelligibility, thereby motivating the development of robust speech enhancement and filtering techniques[1-3].

Classical adaptive filtering methods, such as the Least Mean Squares (LMS) and Normalized LMS (NLMS) algorithms, have long served as the foundation for noise reduction and acoustic echo cancellation systems[4, 5]. These methods, well known for their simplicity and low computational cost, rely on optimizing criteria such as the Mean Square Error (MSE). However, their performance is highly dependent on the characteristics of the noise, the dynamics of the signal, and, most importantly, the adaptation step-size[6, 7]. This dependence leads to a well-known trade-off between fast convergence and low steady-state error, particularly in dispersive or non-stationary acoustic environments.

To overcome these limitations, research has progressively turned toward more advanced approaches, including Blind Source Separation (BSS), Variable Step-Size (VSS) methods [8-10] and, more recently, algorithms based on Artificial Neural Networks (NN) and Deep Learning (DL)[11–16]. These data-driven approaches allow better adaptation to time-varying and spectral variations in the signals, providing more effective separation between the desired speech and interfering noise.

In this context, the work presented in this thesis is part of the broader framework of acoustic noise reduction and speech enhancement using two-sensor adaptive filtering techniques. The main objective is to design and evaluate new architectures and adaptation strategies capable of improving convergence speed, stability, and signal quality, while maintaining low computational complexity for potential real-time implementation.

The first chapter introduces general concepts related to adaptive filtering, including the Wiener filter, LMS and NLMS algorithms, and various noise reduction configurations (single,

two, and multi-sensor systems). This chapter presents the theoretical part required for understanding adaptive filtering techniques.

The second chapter focuses on Blind Source Separation (BSS) methods applied to noise reduction. After presenting the mathematical models of instantaneous and convolutive mixtures, two main separation structures are studied: the Forward and Backward structures. Several two-sensor adaptive algorithms are developed notably Two-sensors LMS, Two-sensors NLMS, and SAD (Symmetric Adaptive Decorrelation) to demonstrate their effectiveness in noise reduction and speech quality enhancement.

The third chapter introduces a novel Neural Network-based Variable Step-Size Feed-forward NLMS (NN-V-FNLMS) algorithm. This method integrates a dynamic, intelligent step-size estimation using a simple neural network to overcome the traditional compromise between fast convergence and low residual error. Additionally, a Voice Activity Detector (VAD) is incorporated to control the filter updates during the silence or noise-only periods, thereby optimizing the computational efficiency and adaptation process.

The fourth chapter extends the last approach with an advanced Deep Learning-based model, noted:DL-VSS-FNLMS. This algorithm employs a Recurrent Neural Network (RNN) with stacked Long Short-Term Memory (LSTM) layers to predict the step-size more accurately and robustly, based on acoustic features such as MFCCs, GTCCs, and ERB/Bark/Mel spectral bands extracted from noisy speech signals. Experimental results confirm that the proposed method significantly improves convergence behavior, noise reduction, and speech quality compared with classical versions.

The final part concludes the thesis by summarizing the major findings and contributions and also outlines several perspectives for future research.

Chapter 1

Generalities on Adaptive Filtering And Noise Reduction Techniques

1.1. Introduction

This first chapter provides the theoretical and conceptual foundations of adaptive filtering and its application to acoustic noise reduction. We begin by introducing essential notions related to speech signals and noise characteristics, followed by an explanation of the acoustic impulse response, which models the behavior of sound propagation in enclosed environments. Subsequently, we discuss the Wiener optimal filter, which serves as a reference framework in signal estimation and noise reduction, derived from the mean square error (MSE) minimization criterion.

The second part of this chapter focuses on adaptive filtering, highlighting its basic principles, structures, and optimization criteria. Special attention is given to stochastic gradient-based algorithms, particularly the LMS and the NLMS algorithms, which form the foundation of many modern adaptive systems. Finally, a comprehensive overview of noise reduction techniques is presented, distinguishing between single-sensor, two-sensor, and multi-sensor approaches, each offering specific advantages and limitations depending on the acoustic environment.

1.2. Acoustic impulse responses

An acoustic impulse response represents the characteristic behavior of a space or environment when subjected to a short impulsive sound. It captures how sound waves propagate, reflect, and decay within a given space over time. This response integrates several acoustic phenomena, including reflection, diffraction, absorption, and reverberation, that collectively define the sound characteristics of a location.
In acoustical systems, two main types of impulse responses are typically distinguished [17]: dispersive andsparse**.**

- *Dispersive impulse responses* are marked by the spreading of signal components over time, leading to a smeared response. In this case, frequency-dependent delays occur, causing different frequency components to arrive at different times.
- *Sparse impulse responses* are instead characterized by a limited number of significant values, with most values being close to zero [18]. Such responses often arise in scenarios with a clear line-of-sight propagation or minimal reflections [19].

Figure 1.1 illustrates examples of real acoustic dispersive and sparse impulse responses.

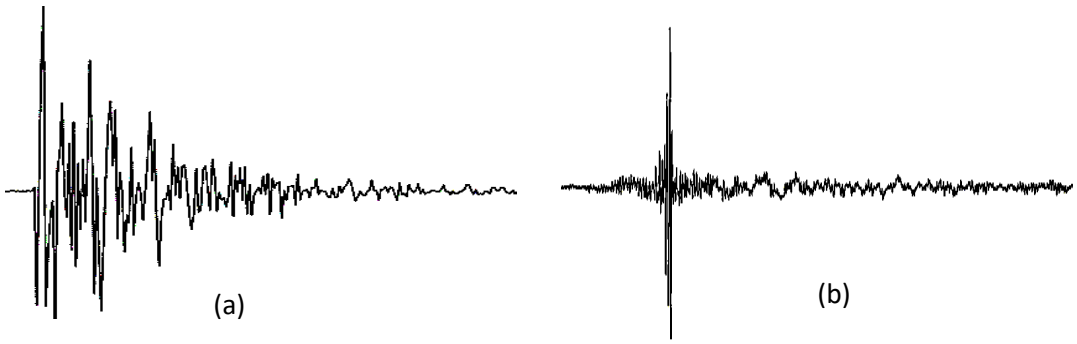

Figure1.1. Real impulse responses, (a) Dispersive, (b) Sparse

1.3. Generalities on the Speech Signal and Noise

Speech is the primary means of communication between humans and representsamajor component of the information transmitted in telecommunication systems. Communication using hands-free systems can be significantly degraded by ambient noise and by reverberation phenomena. For this reason, this section presents general concepts related to the speech signal and the different types of noise.

1.3.1. Speech signal

Speech can be modeled as the result of the excitation of the vocal tract by either a train of impulses or white noise, giving rise respectively to voiced and unvoiced sounds. In the case of voiced sounds, the excitation is a periodic vibration of the vocal folds, caused by the airflow from the respiratory system. This vibratory movement results from successive opening and closing cycles of the glottis. The number of these cycles per second corresponds to the fundamental frequency. As for unvoiced sounds, the airflow passes freely through the glottis without causing vibration of the vocal cords [20].

The speech signal is essentially a continuous phenomenon; periods of silences generally correspond to breathing pauses whose occurrence is random. The speech waveform is neither Gaussian, nor ergodic, nor stationary; however, excellent short-term descriptions of the behavior of the speech production mechanism can be made. A fundamental characteristic of the speech signal is its high degree of redundancy. This redundancy often to an overestimation of the order of the classical autoregressive (AR) model used to represent it the speech signal. This is explained by the fact that the speech signal can be synthesized by convolving the excitation signal with the impulse response of the AR filter. Although the speech signal is a

non-stationary random process in the long term, but it can be considered stationary within analysis windows of approximately 20 to 30 ms[20].

Speech sounds can be broadly classified into three distinct categories: voiced, unvoiced, and silence.

- A voiced sound is a quasi-periodic signal very rich in harmonics of a fundamental frequency (pitch). This type is generated by an AR filter excited by a train of impulses.
- An unvoiced sound is a signal that does not exhibit a periodic structure. This type is generated by the same AR filter excited by white noise.
- Silence simply refers to intervals when the speech signal is absent.

1.3.2. Noise

Noise is defined as any unwanted signal that superimposes itself on the desired signal at any point in a measurement chain or a transmission system. In other words, noise refers to any disturbing phenomenon that interferes with the perception or interpretation of a desired signal (in our case, speech). In physics, acoustics, and signal processing, although noise is inherently random, it possesses certain statistical, spectral, or spatial characteristics.

Noise can be classified according to the following properties [20]:

- *Structure:* continuous, impulsive, or periodic
- *Interaction type:* additive, multiplicative, or convolutive
- Temporal behavior: stationary or non-stationary
- *Frequency characteristics:* narrowband or wideband
- *Dependence:* correlated or uncorrelated
- *Statistical properties:* dependent or independent
- *Spatial characteristics:* coherent or incoherent

White noise is a stochastic process commonly used to model noise in dynamic systems. It is characterized by a power spectral density (PSD) that remains constant over all frequencies.

$$S_{WW}(f) = \sigma_b^2 \qquad \forall f \tag{1.1}$$

A noise with a constant PSD is said to be *white* by analogy with white light, which contains all wavelengths of visible light. White noise corresponds to a purely theoretical model. In fact, it is physically unrealizable because it contains infinite frequencies with infinite average power [21].

The autocorrelation function of white noise is a Dirac impulse [21]. Indeed, by the inverse Fourier transform:

$$R_{WW}(\tau) = TFI\{S_{WW}(f)\} = TFI\{\sigma_b^2\} \tag{1.2}$$

$$R_{WW}(\tau) = \sigma_b^2\ \delta(\tau) \tag{1.3}$$

When the power spectral density is not constant as a function of frequency, the random signal is then called *colored noise*. In this spectral representation, low-frequency colored noise is sometimes called *pink noise* because it contains only long wavelengths. There exist several types of colored noises, such as pink noise and brown noise.

1.4. Wiener filtering

Wiener's theory was formulated by Wiener [22]. In practice, the Wiener filter is derived from the mean square error (MSE) criterion, and many recently proposed algorithms are directly or indirectly related to this filter.

The Wiener filter plays a key role in several digital signal processing applications such as noise reduction, linear prediction, acoustic echo cancellation, and system identification. The extension of Wiener theory from continuous time to discrete time is straightforward, and it is widely used for implementation in the field of signal processing.

1.4.1. General principle

Wiener filtering is among the most widely used classical denoising methods. It is suitable for situations in which the signal or the noise is stationary. The approach consists of minimizing the mean square error between the desired signal and the output of the linear filter. The statistical representation of the filtering problem is shown in Figure 1.2.

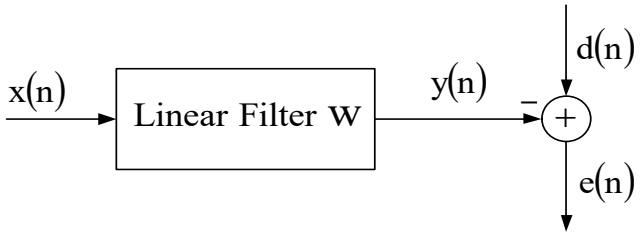

Figure1.2. Statistical representation of the Wiener filtering problem

Thus, the principle of the Wiener filter is to obtain at the output a response as close as possible to a desired response when the input is corrupted by noise. We denote e(n), where e(n) represents the error signal between the desired response d(n) and the filter output y(n); and w(n) represents the vector of the adjustable coefficients of the filter.

To separate a speech signal from noise and reduce the distortion introduced by the signal, the mean square error must be minimized according to the Wiener filter principle.

The output signal of the filter is given by the convolution between the input signal x(n) and the filter coefficients:

$$y(n) = \sum_{m=0}^{M-1} w_m\,(n)\mathrm{x}(n-m) \tag{1.4}$$

It is more convenient to use a vector notation for the filter output, thus equation (1.4) can be written as follows:

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{1.5}$$

Where $\mathbf{w}(n)$ is a vector of length M containing the coefficients of the FIR (Finite Impulse Response) filter, and $\mathbf{x}(n)$ is the vector of the M most recent samples of the input signal. These two vectors are given by:

$$\mathbf{w}(n) = [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T$$

And

$$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-M+1)]^T$$

The error signal can be written as follows:

$$e(n) = d(n) - \sum_{m=0}^{M-1} w_m(n)x(n-m) \tag{1.6}$$

The Wiener filter is the one that minimizes the mean square error (MSE).

$$J = E[e(n)e^*(n)] = E[e(n)^2] \tag{1.7}$$

By introducing the two vectors W(n) and X(n), we obtain:

$$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n) \tag{1.8}$$

Thus, the MSE criterion is given by:

$$J = E\left[\left(d(n) - \mathbf{w}^T(n)\mathbf{x}(n)\right)\left(d^*(n) - \mathbf{w}^T(n)\mathbf{x}(n)\right)\right]$$

$$J = E[(d(n))^2] - \mathbf{w}^T(n)E[\mathbf{x}(n)d(n)^*] - \mathbf{w}^T(n)E[\mathbf{x}^*(n)d(n)] + \mathbf{w}^T(n)E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{w}(n)$$

$$\boldsymbol{J} = E[(d(n))^2] - \mathbf{w}^T(n)E[\mathbf{x}(n)d(n)^*] - \mathbf{w}^T(n)E[\mathbf{x}^*(n)d(n)] + \mathbf{w}^T(n)E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{w}(n) \tag{1.9}$$

1.4.2. Orthogonality

The vector of the optimal filter $W_{opt}$is the one that cancels the gradient of the criterion. We differentiate the MSE criterion with respect to the filter coefficients, and set this derivative equal to zero.

$$\frac{\partial J}{\partial \mathbf{w}(n)} = 0_{Mx1} \tag{1.10}$$

We have, $\frac{\partial J}{\partial \mathbf{w}(n)} = 2E\left\{e(n)\frac{\partial e(n)}{\partial \mathbf{w}(n)}\right\}$

$$\frac{\partial J}{\partial \mathbf{w}(n)} = 2E\{e(n)\mathbf{x}(n)\} \tag{1.11}$$

Therefore, at the optimum, we have:

$$E\{e_{min}(n)\mathbf{x}(n)\} = 0_{M\times 1} \tag{1.12}$$

Where e(n) is the error for which J is minimized (i.e., for the optimal filter). The orthogonality principle means that all the inputs x(n−m), with 0≤m≤M−1, are uncorrelated with the error e(n) [23]. At the optimum, we also have that the error signal e(n) is orthogonal to the filter output y(n).

1.4.3. Wiener–Hopfequation

By expanding equation (1.12), we obtain:

$$E\{\mathbf{x}(n)[d(n)-\mathbf{x}^T(n)\boldsymbol{w}_{opt}]\}=O_{Mx1} \tag{1.13}$$

Let; $E\{\mathbf{x}(n)\mathbf{x}^T(n)\}\mathbf{w}_{opt}=E\{\mathbf{x}(n)d(n)\}$

Or equivalently:

$$\boldsymbol{R}_{xx}\,\mathbf{w}_{opt}=\boldsymbol{r}_{xd} \tag{1.14}$$

Where $\boldsymbol{r}_{xd}$ is the cross-correlation vector between the desired signal d(n) and the input signal x(n).

R_{xx}respresents the autocorrelation matrix of the input signal x(n). This matrix is positive definite, Toeplitz, and Hermitian symmetric.

This last relation is called the Wiener formula or the Wiener–Hopf equation. This solution gives the optimal Wiener filter:

$$\mathbf{w}_{opt}=\boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xd} \tag{1.15}$$

In a complete form, the solution of the Wiener filter can be written as follows:

$$\begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_{M-1} \end{pmatrix}=\begin{pmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(M-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(M-2) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(M-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(M-1) & r_{xx}(M-2) & r_{xx}(M-3) & \cdots & r_{xx}(0) \end{pmatrix}^{-1}.\begin{pmatrix} r_{xd}(0) \\ r_{xd}(1) \\ r_{xd}(2) \\ \vdots \\ r_{xd}(M-1) \end{pmatrix}$$

The Wiener–Hopf equation, which makes it possible to compute the optimal Wiener filter, leads to solving a system of M equations with M unknowns. Figure 1.3 illustrates a Wiener filter represented by a vector of coefficients **w**(n).

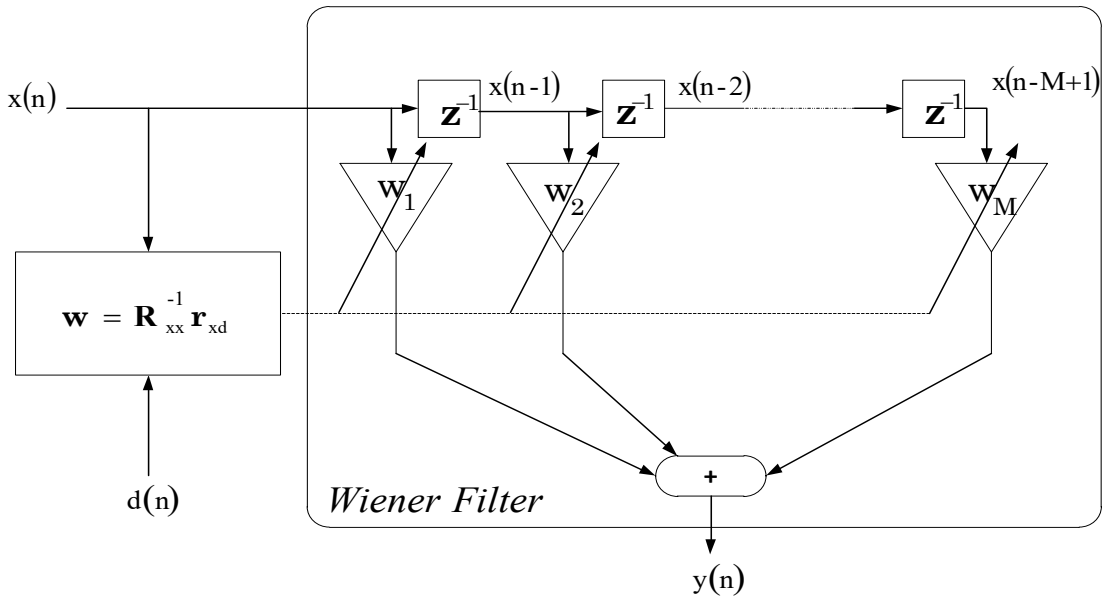

Figure 1.3. Detailed structure of the Wiener filter [24].

It may be preferable to solve this system using an iterative algorithm, particularly recalling that the cost function is quadratic, meaning that the minimum is unique. Adaptive algorithms make it possible to estimate the adaptive filter by using the vector $\mathbf{w}(n)$ of size M, based on a criterion derived from the estimation of the a priori error [25].

1.5. Adaptive filtering

Adaptive filtering is very well known in the field of signal processing. It is used when it is necessary to implement, simulate, or model a system whose characteristics evolve over time. It leads to the implementation of a filter with time-varying coefficients. The variations of the coefficients are defined by optimization criteria and are performed according to adaptation algorithms, which are determined depending on the application. Adaptive filtering is used in many systems, for example, in acoustic echo cancellation, noise reduction, and speech enhancement.

1.5.1. Basic principle

An adaptive filter is a digital filter whose coefficients change automatically depending on external signals. It is used whenever an environment is poorly known or varying, or to suppress disturbances located within the frequency range of the useful signal. The principle of adaptive filtering is shown in Figure 1.4.

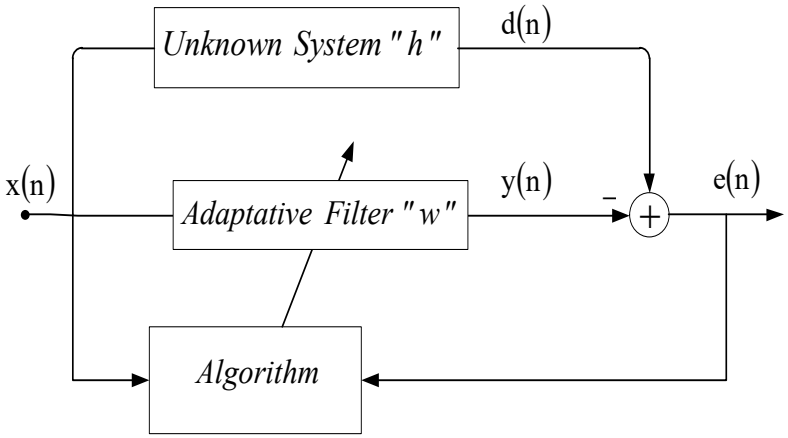

Figure 1.4.Principle of adaptive filtering.

The principle of adaptive filtering consists in processing a received signal in order to provide an output whose difference with a reference signal is minimized. This minimization is achieved by computing the filter coefficients for each pair of input–reference data [26].

We define the different signals used in an adaptive filtering system as follows:

x(n): the input signal of the filter,
w(n): the vector of coefficients of the adaptive filter,
h(n): the vector of coefficients representing the impulse response of the environment (unknown system),
y(n): the output signal of the filter,
d(n): the reference (desired) signal,
e(n): the error signal, defined as the difference between d(n) and y(n).

We assume that the coefficients of the adaptive filter are of thefinite impulse response (FIR) type, represented by the coefficient vector w(n)**.** The use of FIR structures is particularly advantageous because of their inherent stability and well-defined mathematical formulation, which make them widely adopted in adaptive signal processing applications.

Adaptive filters can be classified according to several fundamental aspects:

- **Optimization criterion:** the performance measure that guides the adaptation process, such as the minimization of the mean square error (MSE), the least absolute error (LAE), or more advanced statistical criteria.
- **Update algorithm:** the specific rule used to adjust the filter coefficients, including classical algorithms such as LMS and RLS, as well as their numerous variants and extensions.

- **Programmable filter structure:** the architecture of the filter itself, which can vary from simple tapped-delay-line FIR filters to more complex configurations such as lattice or sub-band adaptive filters.
- **Type of processed signal:** adaptive filters may be designed for single-dimensionalsignals (e.g., audio, communication signals) or multi-dimensional signals (e.g., image or array processing).

This classification highlights the versatility of adaptive filters and emphasizes how their performance strongly depends on the chosen optimization criterion, the adaptation strategy, and the characteristics of the input signals.

Figure 1.5 illustrates a detailed block diagram of adaptive filtering used for system identification.

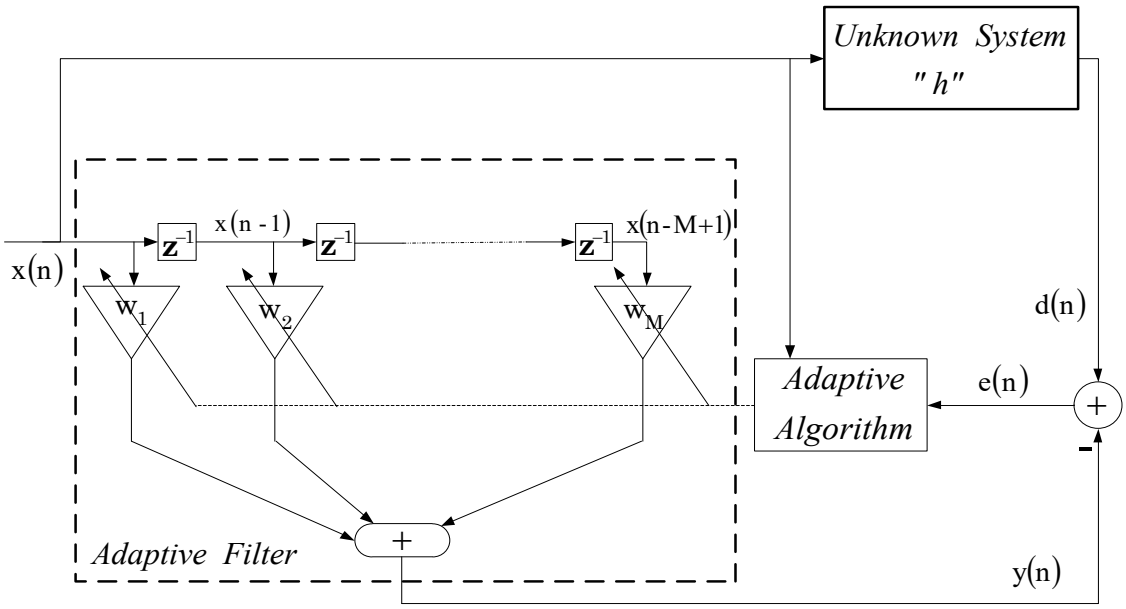

Figure1.5. Detailed block diagram of adaptive filtering [27].

Based on Figure 1.5, the functioning of the adaptive filter can be explained as follows. The input signal x(n) is first convolved with the adaptive filter www, yielding the output signal y(n). This output is subsequently compared with the desired signal d(n), and the error signal e(n) is obtained as the difference between d(n) and y(n).

The error signal e(n) plays a central role in the adaptation process, as it is used to update the coefficients of the adaptive filter. Specifically, at each iteration, the filter coefficients are

modified in proportion to the error signal, according to a predefined minimization criterion. The objective of this iterative adjustment is to progressively reduce the difference between the desired signal d(n) and the filter output y(n).

As the adaptation progresses, the error signal typically decreases and, under favorable conditions, may eventually reach zero. In such cases, the adaptive filter achieves convergence, and its coefficients approximate those of the true impulse response of the unknown system. This convergence property illustrates the fundamental capacity of adaptive filters to identify and track system dynamics in real-time.

1.5.2. Role of adaptive filtering

The main purpose of adaptive filters is to determine a set of coefficients of a system that evolves over time, or alternatively, to adjust the parameter w for a specific objective (e.g., minimization of a cost function J).

In many cases, the cost function J is defined as a function of the input, reference, and output signals of the adaptive filter [27]. Hence, the cost function is strongly related to these three signals, i.e,$f(x(n), d(n), y(n))$

1.5.3. Comparison criteria and selection of adaptive algorithms

Anadaptive algorithm refers to the iterative procedure employed to adjust the coefficients of an adaptive filter with the objective of minimizing a prescribed performance criterion. The efficiency and suitability of the algorithm are determined by three fundamental factors:

- **The definition of the search strategy (or minimization algorithm):** this specifies the approach by which the coefficient space is explored and updated, directly influencing convergence speed and computational complexity.
- **The cost function to be minimized:** this performance measure, such as the mean square error (MSE) or other error-based criteria, defines the optimization objective and governs the adaptation process.
- **The nature of the error signal:** the statistical and structural properties of the error signal strongly affect the algorithm's robustness, accuracy, and convergence behavior.

Taken together, these factors dictate the performance of the adaptive algorithm and its ability to achieve reliable adaptation across a wide range of signal processing applications.

Table 1.1. Comparisoncriteria of adaptive algorithms [31]

Criteria	Definitions
Convergence Rate	Number of iterations required to converge sufficiently close to the Wiener solution.
Misadjustment	The difference between the ensemble average of the squared error and the minimum squared error obtained with Wiener.
Robustness	Resistance to poor conditioning of the data.
Complexity	Hardware aspect, complexity of hardware implementation.
Structure	Hardware aspect, complexity of hardware implementation.
NumericalStability	Influence of quantization errors, problem of error propagation.

1.5.4. Applications

In this section, we discuss some possible choices for the input and reference signals and how these choices are related to the applications. We study a few applications of adaptive filtering such as system identification, inverse modeling, interference cancellation, and prediction [27].

(i) In system identification, the desired signal d(n) is the output of the unknown system excited by a full-band signal, in many cases a white noise signal. The signal x(n) is applied to the adaptive filter (see Figure 1.6). When the output mean square error is minimized, the adaptive filter represents a model of the unknown system.

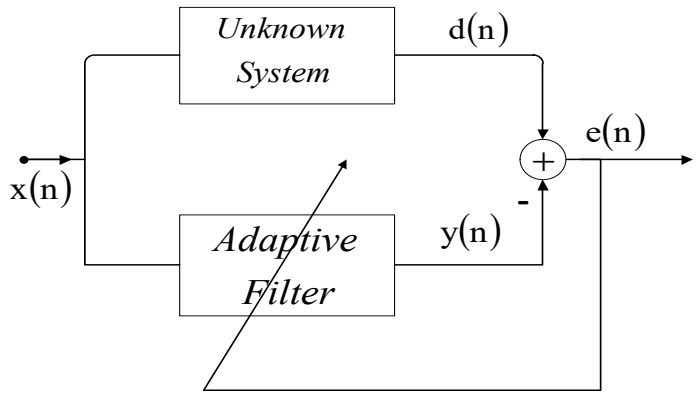

Figure 1.6. System identification using adaptive filtering[27].

(ii) In Figure 1.7, an application of adaptive filtering for channel inverse modeling is illustrated. The full-band input signal x(n) excites the channel input, while the corresponding output of the channel is simultaneously applied to the adaptive filter. The desired signal d(n) is defined as the delayed version of the channel input, which serves as the reference to be recovered through the inversion process.

In the absence of noise, the minimization of the MSE ensures that the adaptive filter converges to the inverse model of the channel. This configuration demonstrates the capability of adaptive filters to reconstruct or compensate for channel distortions, thereby achieving system identification and inversion in real time.

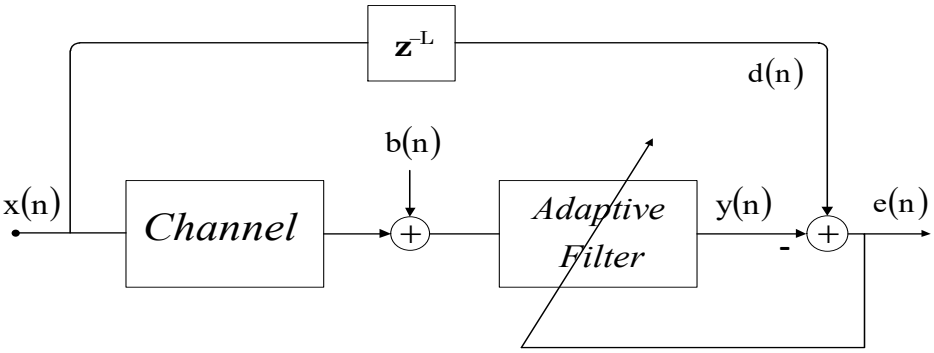

Figure 1.7. Inverse modeling of a channel using adaptive filtering [27].

(iii) In the context of signal enhancement, illustrated in Figure 1.8, the desired signal d(n) corresponds to a signal corrupted by additive noise v(n), while x(n) denotes an available measurable reference signal. By feeding x(n) into the adaptive filter and using the noise-corrupted signal as the desired response, the adaptive process iteratively adjusts the filter coefficients. After convergence, the filter output y(n) provides an enhanced estimate of the original signal, with the noise component significantly attenuated.

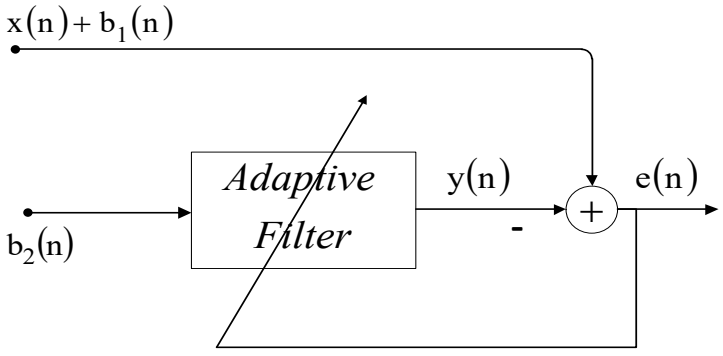

Figure 1.8. Signal enhancement using adaptive filtering [40].

(iv) Finally, we consider the case of prediction (see Figure 1.9), where the desired signal d(n)corresponds to the value of the signal at the current instant n, and the output y(n) represents the prediction obtained from the past samples of the input signal [d(n−1), d(n−2),…]. Through the adaptation process, the filter coefficients are iteratively adjusted so that, after convergence, the adaptive filter provides an accurate model of the input signal. Consequently, the filter can be effectively employed as a predictive model for forecasting future values of the signal [27].

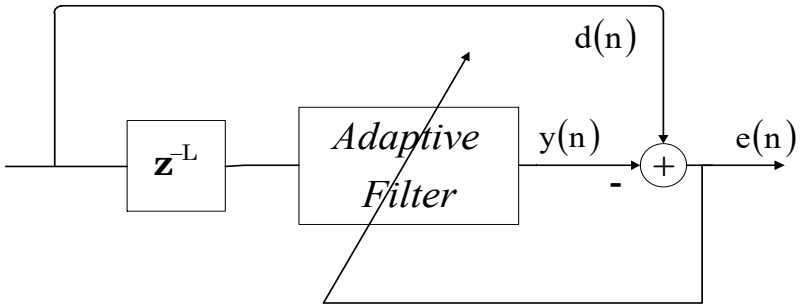

Figure 1.9. Prediction using adaptive filtering.

1.6. Presentation of adaptive algorithms

In Figure 1.10, adaptive filtering algorithms update the coefficients of the filter w(n) so as to minimize the difference between the desired signal d(n) and the filter output y(n), according to a predefined statistical criterion. In general, a basic adaptation algorithm can be expressed in the following vector form:

$$\begin{pmatrix}\text{New filter}\\ \text{coefficients vector}\end{pmatrix} = \begin{pmatrix}\text{Old filter coefficients}\\ \text{vector}\end{pmatrix} + (\text{Step size}) \times (\text{correction term})$$

The definition of the error signal is a critical aspect in the design of adaptive algorithms, as it directly influences several important characteristics such as convergence speed, stability, and computational complexity. Together, the minimization algorithm, the cost function, and the error signal provide the fundamental framework for interpreting, analyzing, and studying adaptive algorithms.

In this section, we introduce some of the most widely used adaptive algorithms in adaptive signal processing, with a particular focus on stochastic gradient-based algorithms, such as the LMS and NLMS algorithms. The stochastic gradient algorithm can be viewed as an approximation of the deterministic gradient descent method.

1.6.1. Stochastic gradient LMS algorithm

The LMSalgorithm, originally introduced by Widrow and Hoff in the early 1960s [28], remains the most popular adaptive algorithm due to itssimplicity, robustness, and lowcomputationalcomplexity. The LMS algorithm is based on a simple stochastic estimate of the gradient, avoiding the need for explicit computation of second-order statistical quantities. One of the key characteristics of the LMS algorithm is that its convergence rate depends not only on the length of the adaptive filter but also on the correlation properties of the filter input signal [28,29].

In the context of the Wiener solution, the autocorrelationmatrixR_{xx} and the cross-correlationvectorr_{xd} are defined as deterministic quantities. In practice, however, these quantities are replaced by their time-varying estimates, denoted $\boldsymbol{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$and $\mathbf{r}_{\boldsymbol{xd}} = \boldsymbol{E}\{\mathbf{x}(n)d(n)\}$at iteration n. For the LMS algorithm, the simplest possible estimates are employed, which are defined as follows:

$$\widetilde{\mathbf{R}}_{xx} = \mathbf{x}(n)\mathbf{x}^T(n) \tag{1.16}$$

$$\tilde{\boldsymbol{r}}_{xd} = \mathbf{x}(n)d(n) \tag{1.17}$$

These are simply the instantaneous estimates of the correlations; therefore, the update formula for the adaptive filter coefficients is given by:

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu\left[\tilde{r}(n) - \widetilde{\mathbf{R}}(n)\mathbf{w}(n)\right]$$

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu\,\mathbf{x}(n)[d(n) - \mathbf{x}^T(n)\mathbf{w}(n)]$$

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu\mathbf{x}(n)e(n) \tag{1.18}$$

The equation (1.18) represents the update formula of the adaptive filter based on the LMS stochastic gradient algorithm. The coefficient vector w(n) is considered a random variable, since at each iteration n it depends on the random processes x(n) and d(n). The parameter μ\muμ denotes the adaptation step size of the LMS algorithm, with the initialization typically chosen arbitrarily at $\mathbf{w}(0)$. The LMS algorithm is particularly attractive due to its computational simplicity. Specifically, it requires only 2M+1 multiplications and 2M additions per iteration, where M is the number of coefficients in the adaptive filter. This low complexity has contributed significantly to its popularity in real-time adaptive signal processing applications.

However, to guarantee the proper operation and convergence of the LMS algorithm, a necessary and sufficient condition must be satisfied [27-30]:

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

Here, λ_{max} denotes the largesteigenvalue of the input signal autocorrelation matrix R_{xx}. A more refined analysis, conducted in the mean-squaresense, but relying on certain debatable assumptions, yields a more restrictive stability condition: $0 < \mu < \frac{1}{\text{Trace}(R)} = \frac{1}{M\sigma_x^2}$

The trace of $\boldsymbol{R}_{xx}$, denoted as tr($\boldsymbol{R}_{xx}$), corresponds to the sum of the diagonal elements of the autocorrelation matrix, which is equal to the sum of its eigen values. On the other hand, $\|\mathbf{x}(n)\|^2$ represents the instantaneousenergy of the input signal x(n).

Algorithm 1.2.LMS Algorithm

Parameters and variables:

M: the size of the adaptive filter $\mathbf{w}(n)$

Step size $0 < \mu < \frac{2}{\lambda_{max}}$

$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-M+1)]$

for i = 1, 2, 3, …

Error signal estimation

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

Update equation

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

end

1.6.2. Normalized LMS algorithm (NLMS)

The LMS algorithm, while very simple to implement, exhibits limitations in the context of acoustic echo cancellation, primarily due to the significant and abrupt variations in the energy of speech signals. These energy fluctuations often cause instability, leading the adaptive filter coefficients to diverge.

To address this issue, Haykin introduced the NLMS algorithm, which modifies the standard LMS by normalizing the coefficient update with respect to the instantaneous energy of the input signal. This normalization ensures that the adaptation step size becomes inversely

proportional to the energy of the input, thereby preventing divergence and improving stability under large signal fluctuations.

In other words, the NLMS algorithm is derived from the LMS algorithm by replacing the fixed step size μ\muμ with a time-varying step size, defined at each iteration as follows [31]:

$$\mu = \frac{\mu_n}{\|\mathbf{x}(n)\|^2} \tag{1.19}$$

The update formula for the adaptive filter coefficients using the NLMS algorithm is the same as that of LMS, except that the step size is normalized by the energy of the input signal.

Thus, the update equation is given by the following formula:

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \frac{\mu_n}{\|\mathbf{x}(n)\|^2 + \varepsilon}\mathbf{x}(n)e(n) \tag{1.20}$$

Where ε is a factor that allows one to follow more or less quickly the variations of energy in the input signal.The convergence of this algorithm is guaranteed for a step size bounded between 0 and 2 [31], i.e., 0 <μ< 2. The details of the NLMS algorithm are given in Table 1.3.

Algorithm 1.3.NLMS Algorithm

Parameters and variables:

M: the size of the adaptive filter w (n)

μ_n : Step size, $0 < \mu_n < 2$

ϵ: small positive constant

$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-M+1)]$

for i = 1, 2, 3, …

Error signal estimation

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

Update equation

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \frac{\mu_n}{\|\mathbf{x}(n)\|^2 + \varepsilon}\mathbf{x}(n)e(n)$$

end

The main advantage of the NLMS algorithm over the classical LMS algorithm lies in its independence from the variance of the input signal. This normalization stabilizes the adaptation process regardless of the input signal's power fluctuations. However, it is important to note that the distribution of the eigenvalues of the input autocorrelation matrix remains unchanged. Consequently, both algorithms exhibit the same dependence of their convergence behavior on the input signal statistics [32],whether the signals are stationary (e.g., white noise) or non-stationary (e.g., speech).

Although the NLMS algorithm introduces additional computational operations compared to the LMS, it still remains among the simplest adaptive algorithms to implement. Due to its balance between stability, robustness, and low complexity, the NLMS is widely employed in practical applications such as acoustic echo cancellation and speech enhancement through noise reduction.

It is possible to improve the performance (convergence) of the NLMS algorithm by modifying the adaptation direction of the adaptive filter coefficients. We can use the family of Affine Projection Algorithms (APA), which are obtained by a multiple-order projection P [33].

1.7. Noise reduction

Noise reduction is a very important step in telecommunication systems. Several structures and algorithms have been proposed to improve communication conditions, i.e., by transmitting the least noisy speech signal possible. The goal is therefore to extract a useful signal (speech) from noisy observations (measured at the microphones). This represents a classical problem in the field of signal processing.

The methods implemented, called denoising methods, and more specifically for our objective, speech enhancement methods, take advantage of all or part of the available information about the nature of the useful signal, the nature of the disturbance, or the properties of the mixture. In the following of this chapter, we will present single-sensor, dual-sensor, and multi-sensor speech denoising methods.

1.7.1. Single-sensor methods

In this part, we analyze the oldest technique for noise cancellation, called the single-sensor method, where only one sensor is used. The goal is to improve listening quality through this method for applications such as mobile telephony and hands-free telephony. These

denoisingmethods can also be used in many other applications. We consider the denoising problem given in Figure 1.10.

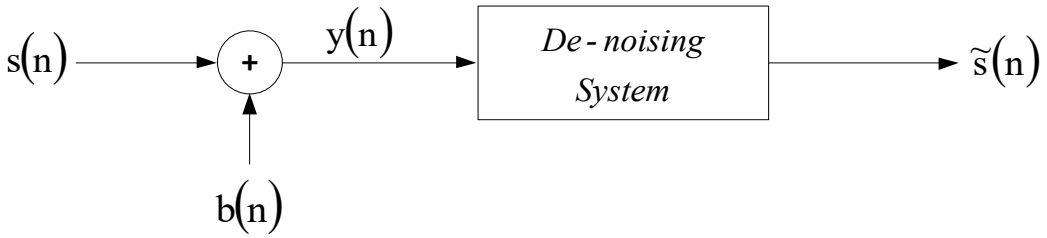

Figure 1.10. Principle of Single-Sensor Denoising

Most single-sensor denoising techniques consist of performing filtering in the frequency domain of the microphone signal.

The filtering attenuates the amplitude of each spectral component of the noisy signal as a function of the estimated signal-to-noise ratio of that component. The methods are differentiated according to the required attenuation, the method of estimating the noise level, and the speech level for each spectral component.

These techniques rely on several fundamental assumptions: the noise and the useful signal are considered uncorrelated within the analysis frame duration, the useful speech signal is assumed to be intermittent, and the human auditory system is largely insensitive to the phase of the signal.Based on these assumptions, the techniques can be broadly classified into three main categories:

- Spectral Subtraction of Power (SSP)
- Spectral Subtraction of Amplitude (SSA)
- Direct implementation of the Wiener solutionthrough open-loop filtering of themicrophone signal, which aims at minimizing the mean squareerror (MSE)

From the model represented in Figure 1.10, we have:

$$y_k(n) = s_k(n) + b_k(n) \tag{1.21}$$

Where κ is the index of the current frame, which contains the same number of samples T. The objective is to restore the clean signal $s_k(n)$ from the observed signal $y_k(n)$

The discrete Fourier transforms (DFT) of the signals $y_k(n)$,$s_k(n)$, and $b_k(n)$are denoted respectively as Y(*f,k*), S(*f,k*) and B(*f,k*), where f represents the frequency. Thus:

$$Y(f,k) = S(f,k) + B(f,k) \tag{1.22}$$

The power spectral densities of the signal can be defined by the following relation:

$$\gamma_y(f,k) = \gamma_s(f,k) + \gamma_b(f,k) \tag{1.23}$$

Where $\gamma_y(f,k)$, $\gamma_s(f,k) et \gamma_b(f,k)$represent respectively the power spectral densities (PSD) of the signals $S_k(n)$, $Y_k(n)$, and $b_k(n)$.

The time-domain signal y(t) can be represented in the frequency domain by its magnitude $|Y(f,k)|$and its phase $|\Phi(f,k)|$.

This latter notation is justified under the assumption of non-correlation between the noise and the signal.

In the frequency domain, equation (1.23) can thus be expressed as:

$$|Y(f,k)|.e^{j\Phi_{y(f,k)}} = |S(f,k)|.e^{j\Phi_{s(f,k)}} + |B(f,k)|e^{j\Phi_{b(f,k)}} \tag{1.24}$$

Thus, the spectrum of the enhanced signal $\tilde{s}(n)$ is expressed by the following relation:

$$\tilde{s}(f,k) = |\tilde{s}(f,k)|.e^{j\Phi_{s(f,k)}} \tag{1.25}$$

The different steps of the speech enhancement process by short-term spectral attenuation are shown in the diagram represented in Figure 1.11.

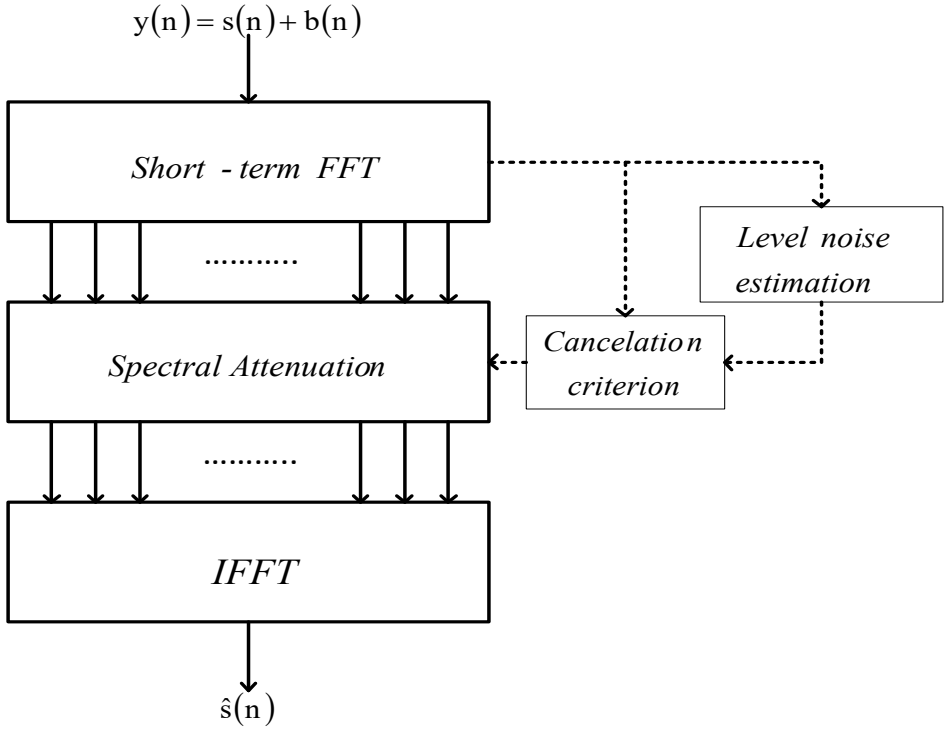

Figure 1.11. General diagram of a spectral attenuation-based denoising method

1.7.1.1. Spectral subtraction

Spectral subtraction relies on the a priori estimation of noise, which is assumed to be additive and either stationary or slowly varying. Under this assumption, noise can be estimated during silent segments of the speech signal. Two basic forms of spectral subtraction exist, distinguished by whether the subtraction is performed on the power spectrum or on the amplitude spectrum**.**

It is important to note, however, that speech signals are inherently non-stationary. They can only be regarded as quasi-stationary over short durations, typically less than 30 ms.Consequently, spectral densities must be estimated using short-term amplitude spectra.

A limitation of the classical formulation arises because the second term of equation (1.25) may become negative. To address this, it can either be set to zero or its sign can be changed, as described in equation (1.26). This adjustment represents one of the earliest improvements to the spectral subtraction technique.

$$|\tilde{s}(f,k)|^2 = \begin{cases} |Y(f,k)|^2 - \left|\tilde{B}(f,k)\right|^2 & if \quad |Y(f,k)|^2 > \left|\tilde{B}(f,k)\right|^2 \\ 0 & else \end{cases} \tag{1.26}$$

The passage to the time domain is carried out by the inverse Fourier transform while keeping the phase of the noisy signal, since the human ear is not very sensitive to phase variations.

$$\tilde{s}(n) = TF^{-1}\left[|\tilde{s}(f,k)|e^{j\varphi_y(f,k)}\right] \tag{1.27}$$

Spectral subtraction algorithms can also be studied from another perspective, that of filtering the observed signal while still relying on an estimation of the noise.

1.7.1.2. Wiener filtering

The Wiener filter is among the most effective classical denoising methods for speech enhancement. It is the estimator $\tilde{S}(f)$ that minimizes the mean square error (MSE) between the input signal and the output signal.

$$E[|\varepsilon(f,k)|^2] = E[|s(f,k) - \tilde{s}(f,k)|^2]$$

$$E[|\varepsilon(f,k)|^2] = E[|S(f,k) - W(f,k)Y(f,k)|^2] \quad (1.28)$$

The expression of the filter is given by equation (1.29):

$$W(f,k) = argminE[|S(f,k) - W(f,k)Y(f,k)|^2] \quad (1.29)$$

According to the projection theorem, there is only one solution for the previous equation. It is given by the orthogonality principle via equation (1.30):

$$E[\varepsilon(f,k)Y(f,k)] = 0 \quad (1.30)$$

The Wiener filter H(f) is expressed in terms of the power spectral densities of speech and noise as follows:

$$W(f,k) = \frac{\gamma_s(f,k)}{\gamma_s(f,k) + \gamma_b(f,k)} \quad (1.31)$$

The estimated speech signal at the output is given by the following linear relation:

$$\tilde{s}(f,k) = W(f,k)Y(f,k) \quad (1.32)$$

In the single-sensor speech denoising problem, only the observed noisy signal is accessible, whereas the previous formulation relies on a priori quantities. One common approach to estimate the power spectral density (PSD) of the clean speech signal consists of iteratively computing the Wiener filter by employing an LPC (Linear Predictive Coding) model of speech. In this framework, the PSD is updated at each iteration from the autoregressive (AR) coefficients [20].

It is worth noting that these three denoising techniques are often accompanied by an overestimation of the noise power, with the objective of achieving the lowest possible residual noise level at the output of the enhancement process. Alternatively, some methods employ a nonlinear spectral subtraction, where the degree of overestimation is adaptively controlled as a function of the signal-to-noise ratio (SNR) at each frequency bin [35-37].

In [38], a well-known method proposed by Ephraim and Malahis based on an amplitude estimator of the speech spectral components, which provides a more accurate reconstruction of the clean signal in the spectral domain.

1.7.2. Two-sensor methods

The principle of dual-sensor adaptive noise cancellation goes back to the contributions of Howells [39] in the late 1950s. The development of the LMS algorithm by Widrow and Hoff [28] enabled the elaboration of this technique in its present form.

Some methods are based on the use of dual-sensor techniques. The environment is considered as a simple convolutive mixture model with two sources (speech and noise):

- The first microphone captures the speech signal s(n) and the noise b(n) convolved with the impulse response h(n).
- The second sensor captures only the noise (see Figure 1.12).

Several algorithms have been proposed that are based on a single filter with a reference path. The filter update is carried out using the error signal, which equals the difference between the reference signal and the filter output. Various adaptive algorithms such as LMS can be used.

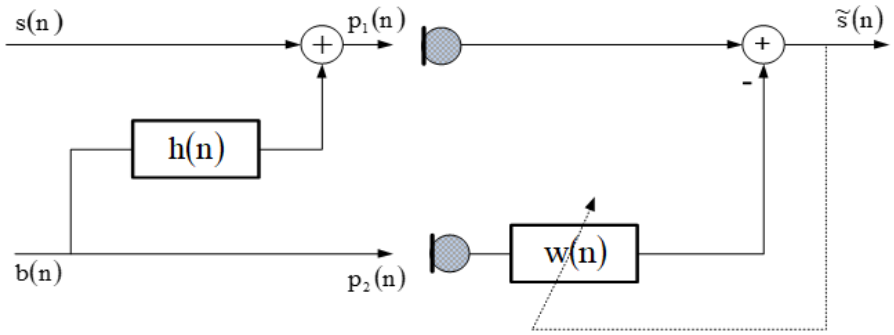

Figure 1.12.Structure of adaptive noise cancellation with reference.

Let us consider a noisy observation $p_1(n)$ of a clean signal s(n), expressed as:

$$p_1(n) = s(n) + \big(b(n) * h(n)\big) \tag{1.33}$$

Where (b(n)∗h(n) denotes the convolution between the noise b(n) and the impulse response h(n).

Now, suppose that at the level of a second source, a noise signal $b_2(n)$ is available, which is highly correlated with b(n) but uncorrelated with the desired signal s(n). The relationship between b(n) and $b_2(n)$ is assumed to be linear, such that:

$$b_2(n) = h(n) * b(n) \tag{1.34}$$

Where $(*)$ denotes the convolution product.

The principle of the Adaptive Noise Cancellation (ANC) technique is to process the reference noise $b_2(n)$ through an adaptive filter in order to obtain the best possible estimate of b(n) in the mean-square error sense. Subtracting this estimate from the noisy observation yields a significant reduction of the noise component at the filter output.

Taking into account the assumed linear relationship between the two noise signals, the output signal of the ANC system can be expressed as:

$$\tilde{s}(n) = p_1(n) - p_2(n) * w(n) \tag{1.35}$$

The two signals $p_1(n)$ and $p_2(n)$ are given by the following relations:

$$p_1(n) = s(n) + h(n) * b(n) \tag{1.36}$$

$$p_2(n) = b(n) \tag{1.37}$$

By substituting these last two equations into equation (1.35), we obtain:

$$\tilde{s}(n) = s(n) + h(n) * b(n) - w(n) * b(n)$$

$$\tilde{s}(n) = s(n) + (h(n) - w(n)) * b(n) \tag{1.38}$$

At the optimum, we have $w_{opt}(n) = h(n)$, and for this equality, the estimated speech signal exactly equals the original signal.

$$\tilde{s}(n) = s(n) \tag{1.39}$$

One way to determine the filter w(n) without any knowledge of either the signal s(n) or the response h(n) is to consider the solution obtained by minimizing the energy of the estimated output signal. Indeed, taking into account the decorrelation between s(n) and b(n), and using equation (1.38)

$$E[\tilde{s}^2(n)] = E[s^2(n)] + E[(h(n) - w(n)) * b(n))^2] \tag{1.40}$$

The lowest value of $E[\tilde{s}^2(n)]$is then reached for $w_{opt}(n) = h(n)$.

The solution to this optimization problem corresponds to the Wiener filter, which is defined as:

$$w_{opt}(z) = \frac{\gamma_{p_1 p_2}(z)}{\gamma_{p_2 p_2}(z)} \tag{1.41}$$

Where the function $\gamma_{p_1 p_2}(z)$represents the cross-spectral density between the two signals $p_1(n)$and $b(n)$, which is defined by:

$$\gamma_{p_1 b}(z) = \gamma_{p_1 p_2}(z) = E[P_1(z)P_2(z^{-1})] \tag{1.42}$$

Where $P_1(z)$ and $P_2(z^{-1})$ represent respectively the z-transforms of $p_1(n)$and $p_2(-n)$.

Another solution to obtain $w_{opt}(n)$ consists in performing an adaptive estimation of the impulse response $h(n)$ parametric modeling of $w(n)$ is then chosen in the form of an FIR filter, whose coefficients $w(n)$ are estimated using an adaptive algorithm. For updating the adaptive filter $w(n)$, several adaptive algorithms can be used, such as stochastic gradient, affine projection, and recursive least squares.

1.7.3. Multi-sensor methods

In this section, a general study of noise reduction using multi-sensor techniques is presented. Several algorithms have been proposed in the time and frequency domains to solve this problem [40, 41]. The main role of multi-sensor techniques in noise reduction is to improve the quality of the estimated speech signal by using multiple sensors.

Let us consider a noise reduction system with “C” sensors in a noisy enclosed acoustic environment, as represented in Figure 1.13. Let us assume that the acoustic environment contains two sources: the speech signals(n), produced by a speaker, and the noise signalb(n). Both signals propagate within the enclosed environment toward the microphones through direct and reflected acoustic paths. After convolution of each source signal with the corresponding impulse response of the environment, thesignals are superimposed at each microphone [42]. The propagation from each source to the microphones is thus fully characterized by its associated acoustic impulse response**.**

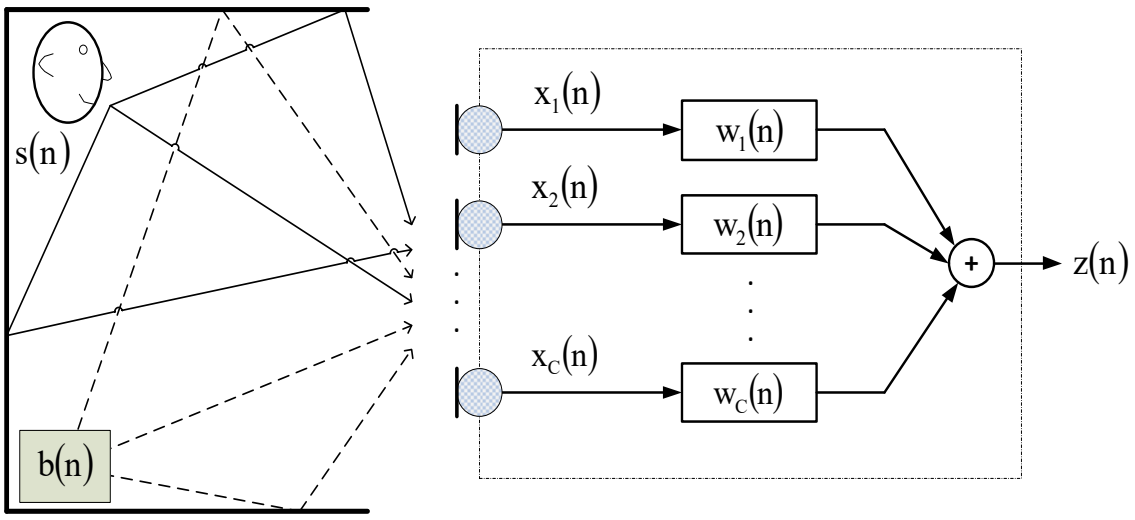

Figure 1.13.Noise reduction using multi-sensor techniques [42].

Each observed signal is composed of two components: one part from the speech signal and the other from the noise, with i = 1, 2, …, C.

$$x_i(n) = s_i(n) + b_i(n) \tag{1.43}$$

And each component of the speech and noise are given respectively by:

$$s_i(n) = s(n) * h_{si}(n) \qquad avec\ i = 1,2,\dots,C \tag{1.44}$$

$$b_i(n) = b(n) * h_{bi}(n) \qquad avec\ i = 1,2,\dots,C \tag{1.45}$$

Where $h_{si}(n)$represents the acoustic impulse response between the speech signal source and the ith microphone. And $h_{bi}(n)$ is the acoustic impulse response from the noise source b(n) to the ith microphone.

Thus, the observed signals $x_i(n)$ can be written by the following equations:

$$\begin{cases} x_1(n) = s(n) * h_{s1}(n) + b(n) * h_{b1} \\ x_2(n) = s(n) * h_{s2}(n) + b(n) * h_{b2} \\ \vdots \\ x_c(n) = s(n) * h_{sc}(n) + b(n) * h_{bc} \end{cases} \tag{1.46}$$

The impulse responses correspond to the coefficients of a finite impulse response (FIR) filter of length M, and are expressed as:

$$h_{si}(n) = \left[h_{si,1}(n) h_{si,2}(n) \dots h_{si,M}(n)\right]^T \ avec\ i = 1,2,\dots,c$$

$$h_{bi}(n) = \left[h_{bi,1}(n) h_{bi,2}(n) \dots h_{bi,M}(n)\right]^T \; avec \; i = 1,2,\dots,c$$

We can write the different signals observed at the level of the C microphones as:

$$\begin{cases} x_1(n) = \sum_{m=1}^{M} h_{s1,m}(n)\, s(n-m) + \sum_{m=1}^{M} h_{b1,m}(n)\, b(n-m) \\ x_2(n) = \sum_{m=1}^{M} h_{s2,m}(n)\, s(n-m) + \sum_{m=1}^{M} h_{b2,m}(n)\, b(n-m) \\ \vdots \\ x_c(n) = \sum_{m=1}^{M} h_{sC,m}(n)\, s(n-m) + \sum_{m=1}^{M} h_{bC,m}(n)\, b(n-m) \end{cases} \quad (1.47)$$

In Figure 1.13, all the observed signals $x_i(n)$ are filtered by the filters $w_i(n)$ and combined to obtain the enhanced speech signal $z(n)$ [42].

$$z(n) = \sum_{i=1}^{c} w_i(n) * x_i(n) \quad (1.48)$$

Where the filters $w_i(n)$ are FIR filters of length M and are given

by: $w_i(n) = \left[w_{i,1}(n), w_{i,2}(n), \dots, w_{i,M}(n)\right]^T aveci = 1,2,\dots,C$

We define the filter w(n) as the combination of all the filters $\mathbf{w}_i(n)$, with i=1,2,…,C

$$W(n) = [w_1^T(n),\; w_2^T(n), \dots, w_c^T(n)]^T$$

In the same way, we define the matrixX(n) of dimension (C×M) as follows:

$X(n) = [x_1^T(n), x_2^T(n), \dots, x_c^T(n)]^T$

$where, \qquad x_i(n) = [x_i(n) x_i(n-1) \dots x_i(n-M+1)]^T$

Thus, the estimated speech signal $z(n)$ can be written in the following form:

$$z(n) = \sum_{i=1}^{c} w_i^T(n) * x_i(n) \quad (1.49)$$

The filters used in multi-sensor noise reduction techniques can be adaptive filters.

1.8. Conclusion

This first chapter established the theoretical basis required to understand the principles of adaptive filtering and its applications in acoustic noise reduction. After introducing the main characteristics of speech signals and acoustic noise, we presented key concepts such as the acoustic impulse response, the Wiener filter, and the MSE criterion that underpins most adaptive algorithms.

The study of adaptive filtering highlighted its importance in systems operating under dynamic and uncertain conditions. The LMS and NLMS algorithms were analyzed as fundamental and widely used adaptive methods, combining simplicity, robustness, and efficiency in real-time applications. Additionally, a review of noise reduction techniques, including single-sensor, dual-sensor, and multi-sensor configurations, illustrated the diversity of existing strategies for enhancing speech quality and intelligibility.

The concepts discussed in this chapter provide the methodological framework upon which the rest of this thesis is built. The next chapter will focus on BSS techniques, which represent a natural extension of adaptive filtering and offer new perspectives for noise cancellation and speech enhancement in reverberant and complex acoustic environments.

Chapter 2

Acoustic Noise Reduction Using Two-Sensor Adaptive Filtering Techniques

2.1. Introduction

Over the past several years, the BSStechniquesbased on two-sensor structure has been the focus of extensive research in diverse fields such as signal processing, telecommunications, biomedicine, and neuroscience. The fundamental objective of BSS is to recover the original source signals from observed mixtures without requiring any prior knowledge of either the sources or the mixing process, apart from the assumption that the sources are mutually statistically independent. The term *"blind"* explicitly highlights this absence of prior information.This problem arises when several signals propagate in an environment, undergo transformations, and combine to form complex mixtures that are observed at the measurement sensors (microphones). Since BSS models many physical situations very well, it has seen developments in numerous application fields.This technique exhibits varying levels of difficulty depending on both the characteristics of the considered sources and the nature of the mixing system. Significant progress has been achieved in the development of such methods, particularly in the domains of acoustic noisereduction and speech enhancement.

Throughout this chapter, we present the main problems related to two-sensor BSS structures. We then introduce the two most commonly used structures (Feed-forward and Feed-back) for addressing this problem in the case of acoustic convolutive mixtures. Finally, we review the different adaptive algorithms proposed in the literature for two-sensor source separation, applied to noise reduction and speech enhancement.

2.2. Problem statement

The problem of blind source separation consists in designing methods capable of characterizing unobservable data, called *"sources"*, based solely on the knowledge of the mixture of this data, called *"*observations [43, 49, 50].

In the following, we present a formulation of this problem, the underlying assumptions, the principle, and some applications of source separation.

2.2.1. Problem

The BSS problem arises in several applications. In this thesis, we focus on its applications within the domain of acoustic noise reduction and speech enhancement.The general class of mixtures takes into account the deformation of the propagated signal and models it by

filtering between the sources and the observations. The general representation of a mixing model is shown inFigure 2.1.

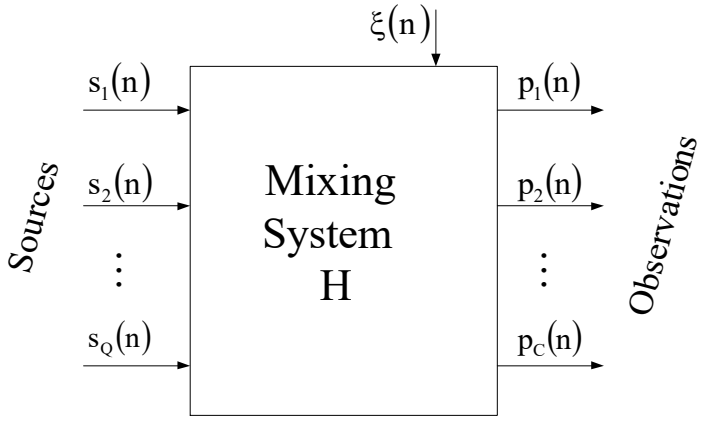

Figure 2.1. General representation of a signal mixture.

In this mixing system, the signals $s_j(n)$ are transformed and then superimposed. At the sensor level, we observe mixed signals $p_i(n)$ resulting from the combination of all the original signals. Furthermore, if we have no prior knowledge of either the original sources $s_j(n)$ or of the mixing system H (the propagation channel), we are faced with a very complex problem.

The Blind Source Separation (BSS) problem can be modeled in a simple way using the following general form:

$$p(n) = H\{s(n)\} + \xi(n) \tag{2.1}$$

Where:

$s(n) = [s_1(n), s_2(n), \dots, s_Q(n)]$represents the vector of the Q source signals

$p(n) = [p_1(n), p_2(n), \dots, p_c(n)]$is the vector of the C mixed (noisy) signals

$H\{.\}$is the mixing system (transfer function).

$\xi(n) = [\xi_1(n), \xi_2(n), \dots, \xi_c(n)]$It is the vector of the C additive disturbing noises.

2.2.2. Assumptions

Separation is said to be blind if it is possible to separate the signals without any prior knowledge of the mixing system and when the sources are unobservable. Without additional assumptions, blind source separation appears as a very complex problem [44]. In this section,

we present a few assumptions about the mixture and the sources in order to solve this problem.

- **Assumption 1:** The number of sources Q equals the number of observations C; in some cases, C<Q
- **Assumption 2:** The sources are statistically independent. From a mathematical perspective, this implies that the joint probability density of the Q sources can be factorized asthe product of their marginal densities.
- **Assumption 3:** Most source separation methods deal with the linear case of the propagation environment.
- **Assumption 4:** The sources are stationary random processes, zero-mean, and of unit variance.
- **Assumption5:** The noise signals in relation (2.1) are considered negligible (zero).

2.2.3.Generalprinciple

Under the assumption of statistical independence of the sources, the blind source separation (BSS) problem is addressed by applying suitable transformations to theobserved mixtures so as to recover signals that are as statistically independent as possible.

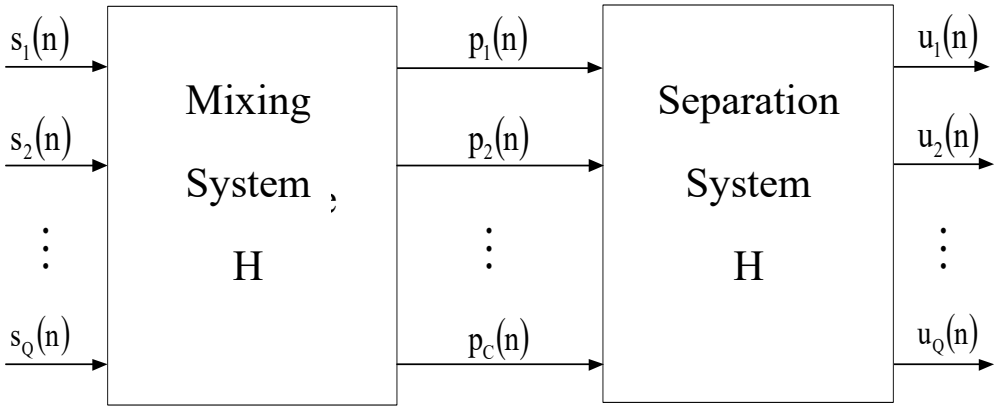

Figure 2.2. General Principle of BSS technique

The main objective of BSS is to estimate the source signals using a separation system W (see Figure 2.2). If we have at least as many sensors as sources (C=Q) and if the structure of the mixing system is known, the problem becomes equivalent to identifying the mixing system H.

The only prior information available is the statistical independence of the sources; therefore, the separating matrix W is estimated in such a way that the components of the vector u(n) become independent. The form of the matrix W depends on the type of model: for an instantaneous linear mixture, it is a matrix with real coefficients, while for a convolutive linear mixture, the coefficients are filters.

2.3.Signalmixing

Signal mixtures can be classified into two main types: instantaneous or convolutive. They may also be time-varying or time-invariant. In the context of this thesis, we will restrict ourselves to time-invariant linear convolutive mixtures between the speech signal and noise.

2.3.1.Instantaneouslinearmixing system

The simplest mixing configuration is the instantaneous linear mixture, in which it is assumed that the source signals arrive simultaneously at all sensors but with different intensities, regardless of the positions of the sources relative to the sensors. The C observations are then expressed as a function of the Q source signals. Figure 2.3 illustrates a model of the instantaneous linear mixture[48].

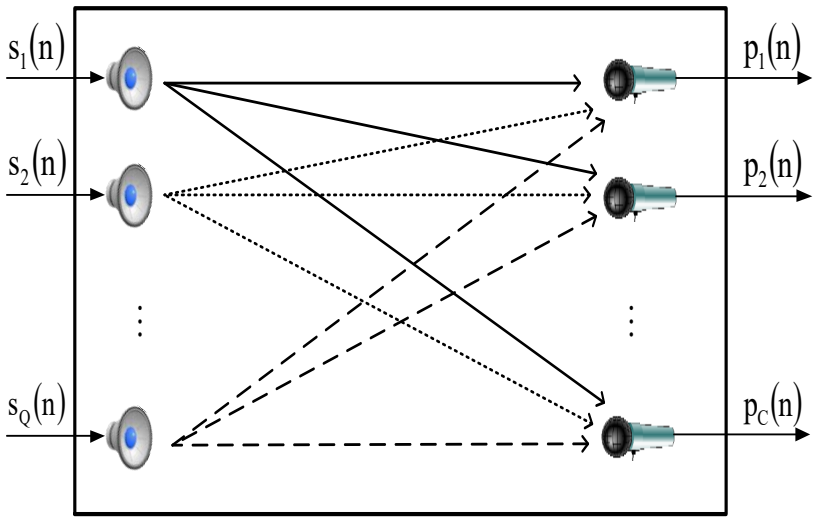

Figure 2.3. Instantaneous linear mixing model with Q sources and C observations.

For this type of mixing, the observations at a given instant *n* are linear combinations of the sources at the same instant *n*.

$$p(n) = \sum_{j=1}^{Q} h_{ji} s_j(n) \qquad i = 1,2,\dots,C \qquad (2.2)$$

Where$\boldsymbol{h_{ji}}$are scalar coefficients of the instantaneous linear mixing process.

An extension of the instantaneous linear mixture is the so-called anechoic mixture, in which the arrival times of the signals at the different sensors are delayed by a certain duration. The delay depends on the position of each source relative to each sensor, i.e., the travel time of a sound wave increases with the distance between the source and the sensor. The relationships between the observations and the sources are given by the following formula[48]:

$$p_i(n) = \sum_{j=1}^{Q} h_{ji} s_j(n - \tau_{ji})\, i = 1,2, \dots, C \qquad (2.3)$$

Whereτ_{ji} represent the delays between the sources and the sensors.

2.3.2.Convolutivelinear mixing system

The convolutive linear mixture is the most complex configuration of linear mixing and the one that most closely resembles real environmental conditions in enclosed spaces. Convolutive mixing can be viewed as an extension of the anechoic mixture, in which multiple propagation paths between the sources and the observations are considered (see Figure 2.4)

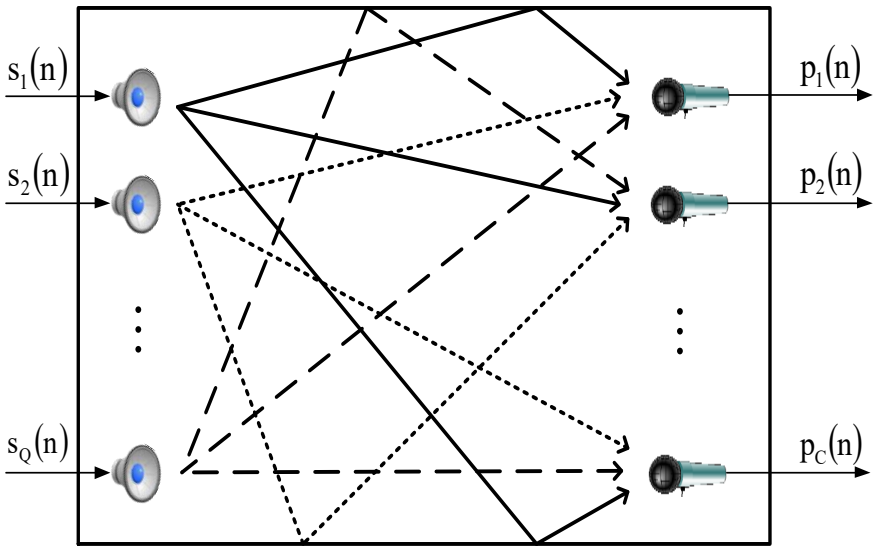

Figure 2.4. Diagram of the convolutive mixing process with Q sources and C observations[48].

In this type of mixing, not only are the transmission delays between the sources and the sensors considered, but also the multiple reflections of the source signals on the walls of an enclosed environment, for example. The different propagation paths depend on the emission

points, the sensor positions, and the geometry of the room. The overall model of the convolutive linear mixing with Q sources and C observations is shown in Figure 2.5 [48].

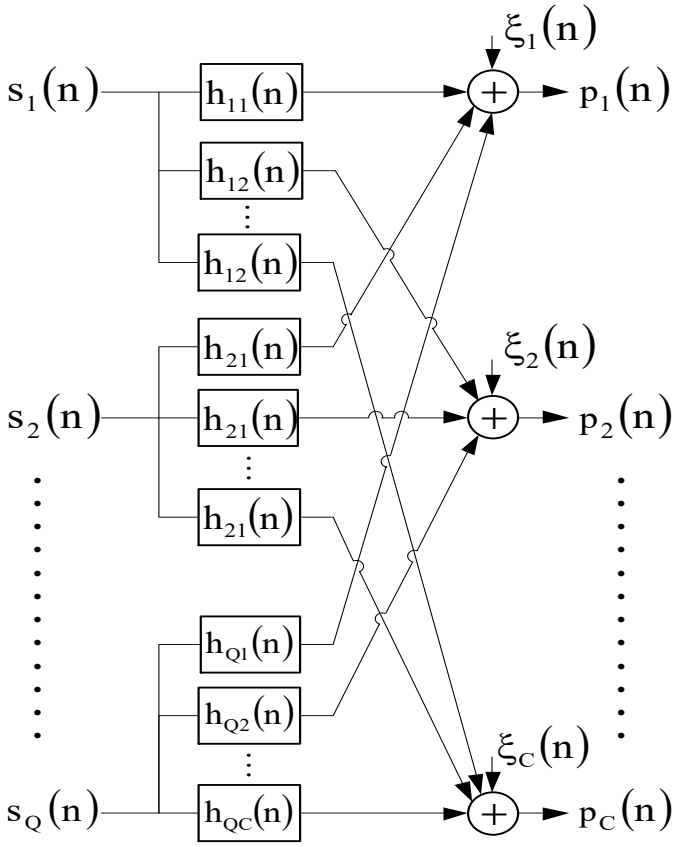

Figure 2.5. Model of the convolutive linear mixing with Q sources and C observations.

In the time domain, the convolutive mixing model is expressed as follows:

$$p_i(n) = \sum_{j=1}^{Q} h_{ji} * s_j(n) + \xi_i(n) \qquad i = 1,2,\dots,C \qquad (2.4)$$

$h_{ji}(n)$represents the impulse response between the j[th] source and the i[th] sensor,

* denotes the convolution product,

$\xi_i(n)$ corresponds to the additive noise at the i[th] microphone.

2.4.Convolutivelinear mixing with two-sensors

In the framework of this thesis, we address the problem of noise cancellation using two-sensor source separation methods. We consider the scenario involving two sources and two

microphones: one corresponding to a speech source (speaker) and the other to a disturbance source (noise).

We assume that this mixture is of the convolutive linear type, combining a speech signal and noise. This model isillustratedin Figure 2.6.

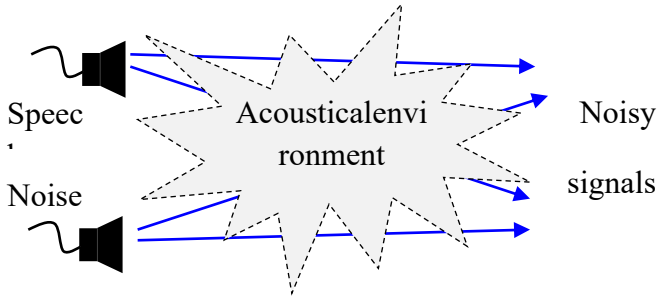

Figure 2.6. Convolutive mixing between the speech signal and the noise.

2.4.1. Complete two-sensor convolutive linear mixing

The convolutive linear mixture between the speech signal and the noise can be represented by a complete model, as shown in Figure 2.7.

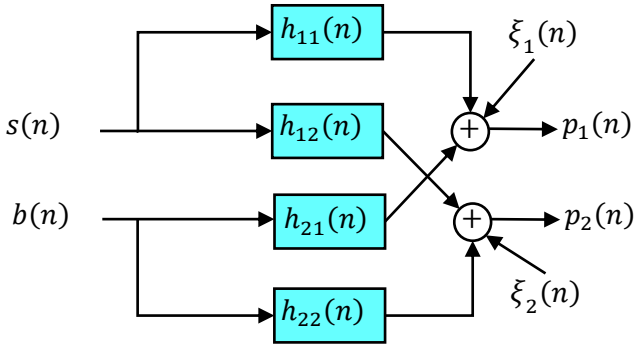

Figure 2.7. Complete structure of a two-sensor convolutive mixture.

At the output of the two microphones, we observe a superposition of the unknown primitive signals according to an unknown mixing process. In general, this corresponds to a full-band convolutive linear mixture of the signals, which depends on the propagation conditions in the environment, the positions of the microphones and sources, and the acoustic characteristics of the medium.

The equations of the observed signals at the output of this mixture are given by:

$$p_1(n) = s(n) * h_{11}(n) + b(n) * h_{21}(n) + \xi_1(n) \quad (2.5)$$

$$p_2(n) = b(n) * h_{22}(n) + s(n) * h_{12}(n) + \xi_2(n) \quad (2.6)$$

$h_{11}(n)$and$h_{22}(n)$ represent the impulse responses of the direct acoustic channels.
$h_{12}(n)$and$h_{21}(n)$account for cross-coupling effects between sources and microphones.
$s(n)$denotes the speech signal**,**
$b(n)$thenoise signal,
$p_1(n)$and$p_2(n)$ respectively represent the two noisy signals,
$\xi_1(n)$and$\xi_2(n)$ correspond to the additive noises at the microphones.

2.4.2. Simplified two-sensor convolutive linear mixing

Under these conditions [45], [51-53], [60], we can adopt a simplified model of the convolutive mixing, in which the two impulse responses $h_{11}(n)$and$h_{22}(n)$areapproximated as all-pass filters[46, 47].Hence, the simplified convolutive mixing model is shown in Figure 2.8.

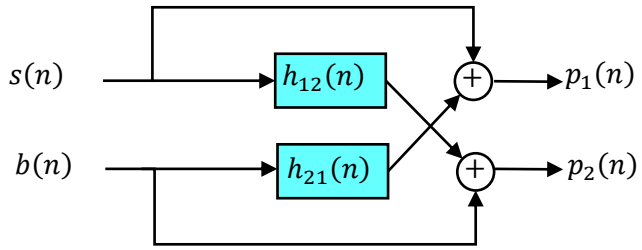

Figure2.8. Simplified structure of a two-sensor convolutive mixture [54], [60].

In Figure 2.8, the two observed signals are given by:

$$p_1(n) = s(n) + b(n) * h_{21}(n) \quad (2.7)$$

$$p_2(n) = b(n) + s(n) * h_{12}(n) \quad (2.8)$$

2.5. Two-sensor source separation structures

BSS is a relatively recent research discipline whose objective is to recover a set of source signals from noisy or mixed observations, which are mixtures of these signals and the impulse responses of the environment.There are two BSS structures: the forward structure and the backward structure, which can be used for acoustic noise cancellation and speech enhancement.

2.5.1. Feed-forward structure

The most widely used structure is the Feed-forward structure, which is shown in Figure 2.9.This structure is employed to estimate the two original signals (with $u_1(n)$ denoting the estimated speech signal) from the observed signals only, without any prior information about either the mixing process or the source signals, relying instead on their statistical independence [45, 17].

In the forward structure, two symmetric adaptive filters,$w_{12}(n)$ and$w_{21}(n)$, are used to identify the two impulse responses$h_{12}(n)$ and$h_{21}(n)$, respectively.The theoretical solution is then given by $w_{12}^{opt}(n) = h_{12}(n)\ and\ w_{21}^{opt}(n) = h_{21}(n)$[47, 48].

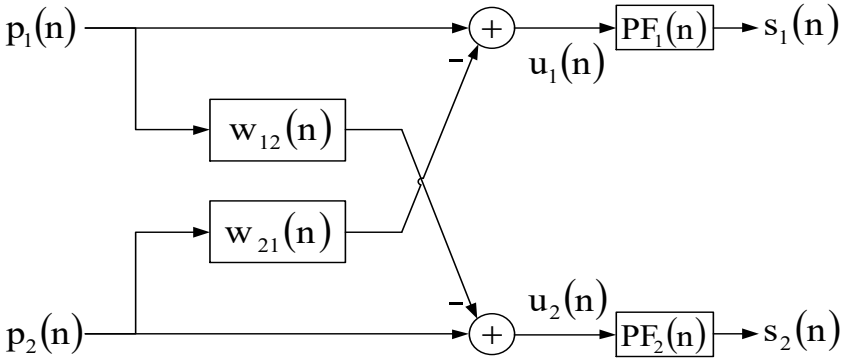

Figure 2.9. Symmetric Forward BSS Structure.

At the output of the forward structure (see Figure 2.9), we obtain the two estimated signals $u_1(n)$ and$u_2(n)$ which are given by the following two formulas:

$$u_1(n) = p_1(n) - p_2(n) * w_{21}(n) \tag{2.9}$$

$$u_2(n) = p_2(n) - p_1(n) * w_{12}(n) \tag{2.10}$$

By substituting equations (2.7) and (2.8) into the two expressions of the estimated signals $u_1(n)1$ and$u_2(n)$, we obtain:

$$u_1(n) = b(n) * [h_{21}(n) - w_{21}(n)] + s(n) * [\delta(n) - h_{12}(n) * w_{21}(n)] \tag{2.11}$$

$$u_2(n) = s(n) * [h_{12}(n) - w_{12}(n)] + b(n) * [\delta(n) - h_{21}(n) * w_{12}(n)] \tag{2.12}$$

By applying the optimality assumption for the two adaptive filters$w_{21}^{opt}(n) = h_{12}(n)$and $w_{21}^{opt}(n) = h_{21}(n)$, we then obtain the outputs$u_1(n)$ and$u_2(n)$ which are given by the following two expressions:

$$u_1(n) = s(n) * [\delta(n) - h_{12}(n) * w_{21}(n)] \tag{2.13}$$

$$u_2(n) = b(n) * [\delta(n) - h_{21}(n) * w_{12}(n)] \tag{2.14}$$

- **Post-filters Insertion**

We can clearly observe that the two output signals,$u_1(n)$and$u_2(n)$, converge respectively toward the two original source signals, $s(n)$ and$b(n)$, though a slight distortion modification.To avoid this distortion, two post-filters $PF_1(n)$ and$PF_2(n)$ can be added at the output of the forward structure, respectively (see Figure 2.9).
These two post-filters are given by:

$$PF_1(n) = PF_2(n) = \frac{1}{\delta(n) - w_{12}(n) * w_{21}(n)} \tag{2.15}$$

At the output of the two post-filters$PF_1(n)$ and$PF_2(n)$, the two signals$s_1(n)$ and$s_2(n)$ converge respectively towards the two original signals$s(n)$ and$b(n)$.

Automatic frequency-domain approaches have been proposed in [47, 48]. These approaches are used to estimate the two post-filters, i.e., to recover the original speech signal and mitigate the distortion present in the two output signals of the forward structure, without requiring explicit post-filtering.

2.5.3. Feed-back structure (Backward)

The second blind source separation technique is the backward structure, which is shown in Figure 2.10.This extension can be considered a very effective structure for noise reduction and speech enhancement[55].

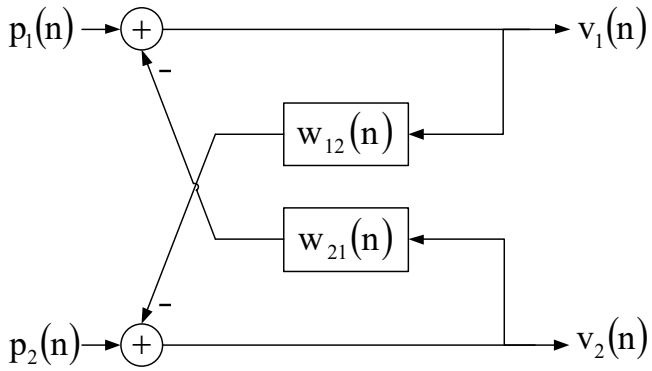

Figure 2.10. Symmetric Feed-back BSS Structure[48]**.**

At the output of this structure, the first estimated signal$v_1(n)$ is obtained as the result of the subtraction between the first mixture signal$p_1(n)$ and the output of the secondfilter$w_{21}(n)$, i.e. by using the second output signal$v_2(n)$ to estimate the signal $v_1(n)$.Thus, the two output signals $v_1(n)$ and $v_2(n)$ of the backward structure are given by the following two equations:

$$v_1(n) = p_1(n) - v_2(n) * w_{21}(n) \tag{2.16}$$

$$v_2(n) = p_2(n) - v_1(n) * w_{12}(n) \tag{2.17}$$

By substituting the formulas$p_1(n)$and $p_2(n)$in the formulas of $v_1(n)$ and$v_2(n)$, we then obtain the following two expressions:

$$v_1(n) = [b(n) * (h_{21}(n) - w_{21}(n)) + s(n) * (\delta(n) - h_{12}(n) * w_{21}(n))] * (\delta(n) - w_{12}(n) * w_{21}(n))^{-1}$$

$$v_2(n) = [s(n) * (h_{12}(n) - w_{12}(n)) + b(n) * (\delta(n) - h_{21}(n) * w_{12}(n))] * (\delta(n) - w_{21}(n) * w_{12}(n))^{-1}$$

By applying the optimality assumption for the two adaptive filters$w_{12}^{opt}(n) = h_{12}$and $w_{21}^{opt}(n) = h_{21}$, we now obtain the following two output signals$v_1(n)$ and$v_2(n)$:

$$v_1(n) = s(n) \tag{2.18}$$

$$v_2(n) = b(n) \tag{2.19}$$

2.6. Two-sensor adaptive filtering algorithms

In the remainder of this chapter, we will present several adaptive filtering algorithms applied to the two-sensor separation structures**.**

2.6.1. Two-sensor LMS algorithm

The LMS algorithm is the most widely used in the field of adaptive filtering. It is among the first adaptive algorithms implemented in blind source separation structures [56].By using the two-sensorLMS algorithm, the two symmetric adaptive filters$\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$ are updated. The application of this algorithm in the forward structure is very straightforward.In Figure 2.11, we present the model of the convolutive mixture and the adaptation of the two separation filters in the forward structure.

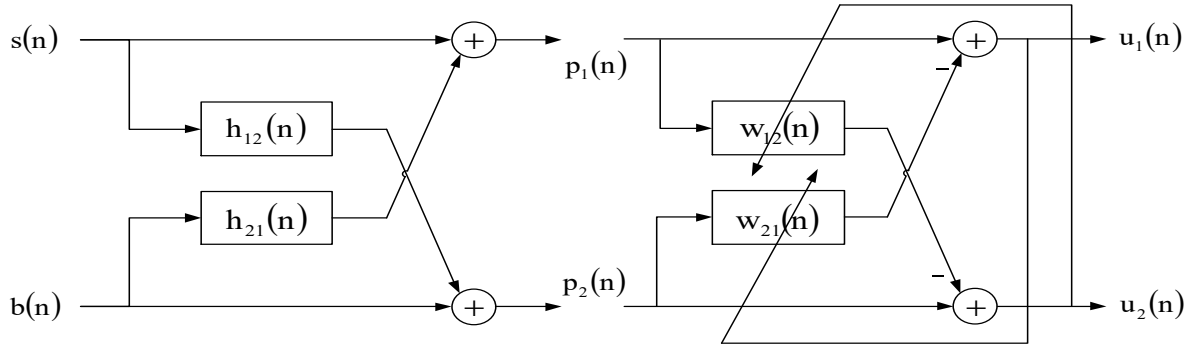

Figure 2.11. Convolutive mixing system and the two-sensor feed-forward structure.

By adapting the update equation of the LMS algorithm to the two adaptive filters of the two-sensor feed-forward structure, we obtain the 2FLMS algorithm (Two-sensors Feed-forward LMS*)*.

The two update equations of the filters$\mathbf{w}_{12}(n)$ and$\mathbf{w}_{21}(n)$ using the 2FLMS algorithm are given by:

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \mu_{12}\,\mathbf{p}_1(n)u_2(n) \tag{2.20}$$

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \mu_{21}\,\boldsymbol{p}_2(n)u_1(n) \tag{2.21}$$

Where μ_{12} and μ_{21} represent the fixed adaptation step-sizes of the two adaptive filters $\mathbf{w}_{12}(n)$and $\mathbf{w}_{21}(n)$, respectively. The vectors $p_1(n)$ and $p_2(n)$ contain the M most recent samples of the mixed signals $p_1(n)$and $p_2(n)$. The necessary and sufficient condition for the convergence of this algorithm is identical to that of the LMS algorithm.

Figure 2.12 illustrates the convolutive mixing model and the implementation of the adaptation process of the two filters in the feed-back structure.

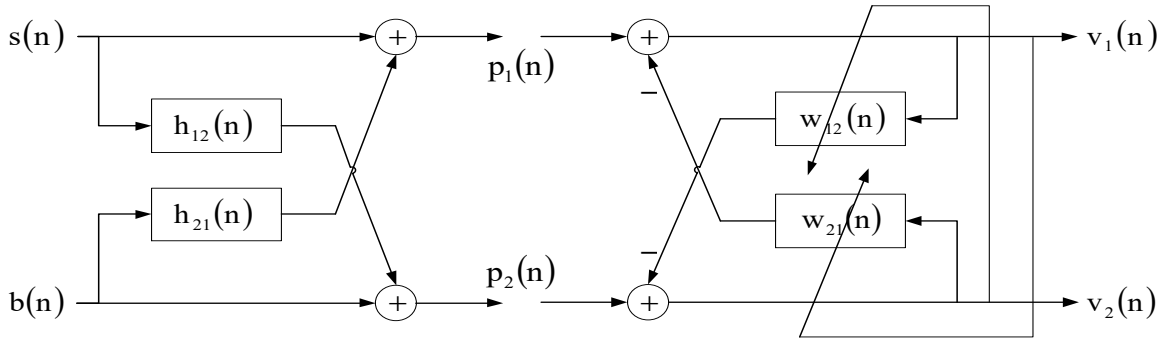

Figure 2.12. Convolutive mixing system and the two-sensor feed-back structure.

By using the LMS algorithm (2BLMS: *Two-sensors Backward LMS*), the two update equations of the filters $\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$can be written in the following form:

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \mu_{12}\,\mathbf{v}_1(n)v_2(n) \tag{2.22}$$

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \mu_{21}\,\mathbf{v}_2(n)v_1(n) \tag{2.23}$$

Where$\mathbf{v}_1(n)$ and$\mathbf{v}_2(n)$ denote the vectors containing the M most recent samples of the output signals$v_1(n)$ and$v_2(n)$, respectively.

Algorithm 2.1. Two-sensors LMS Algorithms (2LMS)[47, 48]

Feed-forward (2FLMS)	**Feed-back (2BLMS)**
$p_1(n) = [p_1(n), p_1(n-1), \dots, p_1(n-M+1)]^T$	$\mathbf{v}_1(n) = [v_1(n), v_1(n-1), \dots, v_1(n-M+1)]^T$
$p_2(n) = [p_2(n), p_2(n-1), \dots, p_2(n-M+1)]^T$	$\mathbf{v}_2(n) = [v_2(n), v_2(n-1), \dots, v_2(n-M+1)]^T$
$for\ n = 0,1,2,3, \dots$	$for\ n = 0,1,2,3, \dots$
Estimation of the output signals	*Estimation of the output signals*
$u_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{p}_2(n)$	$v_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{v}_2(n)$
$u_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{p}_1(n)$	$v_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{v}_1(n)$
Filter update equations	*Filter update equations*
$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \mu_{12}\,u_2(n)\mathbf{p}_1(n)$	$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \mu_{12}\,\mathbf{v}_2(n)v_1(n)$
$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \mu_{21}\,u_1(n)\mathbf{p}_2(n)$	$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \mu_{21}\,\mathbf{v}_1(n)v_2(n)$
End	End
Parameters and variables	
M : Size of the adaptive filters of $\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$	
μ_{12} and μ_{21} : Fixedstep size	

2.6.2. Two-sensorsnormalized LMS algorithm

The NLMS algorithm consists in normalizing the adaptation step size μof the LMS algorithm.Thus, in the two-sensors normalized LMS algorithm (*2NLMS*), the two step sizesμ_{12} and μ_{21} are normalized by the energy of the input signals $p_1(n)$ and$p_2(n)$, respectively.For the implementation of the 2CNLMS algorithm, the two structures represented in Figures 2.11 and 2.12 are used.The NLMS algorithm is applied to update the coefficients of the two filters by using the Feed-forward are given by the next expressions:

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \frac{\mu_{12,n}}{\|\mathbf{p}_1(n)\|^2 + \varepsilon}\mathbf{p}_1(n)u_2(n) \tag{2.24}$$

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \frac{\mu_{21,n}}{\|\mathbf{p}_2(n)\|^2 + \varepsilon}\mathbf{p}_2(n)u_1(n) \tag{2.25}$$

For the Feed-back structure, the updating expressions of the two adaptive filters are given by:

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \frac{\mu_{12,n}}{\|\mathbf{v}_1(n)\|^2 + \varepsilon}\mathbf{v}_1(n)v_2(n) \tag{2.26}$$

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \frac{\mu_{21,n}}{\|\mathbf{v}_2(n)\|^2 + \varepsilon}\mathbf{v}_2(n)v_1(n) \tag{2.27}$$

Where$\varepsilon$is a small positive constant introduced to prevent division by zero.The convergence of this algorithm is guaranteed for$0 < \mu_{12,n} < 2 \;\; and \;\; 0 < \mu_{21,n} < 2$.

Algorithm 2.2 Two-sensors Normalized LMS Algorithms (2NLMS) [48].

2FNLMS	**2BNLMS**
$\mathbf{p}_1(n) = [p_1(n), p_1(n-1), \dots, p_1(n-M+1)]^T$	$\mathbf{v}_1(n) = [v_1(n), v_1(n-1), \dots, v_1(n-M+1)]^T$
$\mathbf{p}_2(n) = [p_2(n), p_2(n-1), \dots, p_2(n-M+1)]^T$	$\mathbf{v}_2(n) = [v_2(n), v_2(n-1), \dots, v_2(n-M+1)]^T$
$for\ n = 0,1,2,3, \dots$	$for\ n = 0,1,2,3, \dots$
Estimation of the output signals	*Estimation of the output signals*
$u_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{p}_2(n)$	$v_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{v}_2(n)$
$u_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{p}_1(n)$	$v_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{v}_1(n)$
Filter update equations	*Filter update equations*
$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \frac{\mu_{12,n}}{\|\mathbf{p}_1(n)\|^2 + \varepsilon}\mathbf{p}_1(n)u_2(n)$	$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) + \frac{\mu_{12,n}}{\|\mathbf{v}_1(n)\|^2 + \varepsilon}\mathbf{v}_1(n)v_2(n)$
$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \frac{\mu_{21,n}}{\|\mathbf{p}_2(n)\|^2 + \varepsilon}\mathbf{p}_2(n)u_1(n)$	$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) + \frac{\mu_{21,n}}{\|\mathbf{v}_2(n)\|^2 + \varepsilon}\mathbf{v}_2(n)v_1(n)$
End	*End*
Parameters and variables M: Adaptive filter length of $\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$ $\mu_{12,n}\ and\ \mu_{21,n}$: Fixed step size, ε: small positive constant	

2.6.3. Symmetric adaptive decorrelation algorithm (SAD)

In this subsection, we discuss a symmetric approach based on the principle of decorrelation, applied to the two-sensor structures, forward and backward [17][45][57].In [58],Theauthors demonstrated that the least-squares criterion is equivalent to the adaptivedecorrelation criterion. In this approach, decorrelation is carried out between an estimate of the speech signal and an estimate of the noise. The least-squares criterion is thus replaced by the decorrelation criterion, and owing to its full symmetry, the algorithm functions as asignalseparator rather than a noise canceller. This method is commonly referred to as the SAD algorithm[58, 59]**.**Considering the two-sensor denoising problem (with reference) shown in Figure 2.13, the Adaptive Decorrelation (AD) algorithm was proposed for acoustic noise cancellation. Assuming that the two signals$s(n)$and$b(n)$ are statistically independent, the cross-correlation product between them is therefore equal to zero, which is the required condition for the operation of the AD algorithm, $C_{sb}(m) = E[s(n)b(n-m)] = 0 \quad \forall m.$

The variances of the speech and noise signals are $\sigma_s^2$and σ_b^2, respectively.

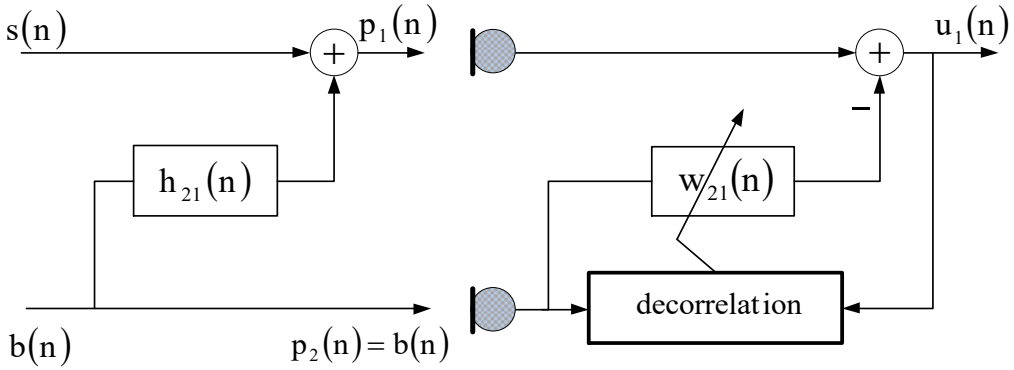

Figure 2.13. Structure of the Adaptive Decorrelation Algorithm [48].

The operation of the AD algorithm is based on the minimization of the error energy. This error minimization is equivalent to the cross-correlation between the estimated signal $u_1(n)$ and the mixture signal $p_2(n)$ [58].

$$\frac{\partial \varepsilon_1(n)}{\partial \mathbf{w}_{21}(m)} = 2C_{u_1 p_2}(m) m = 0,1,\dots,M-1 \tag{2.28}$$

Where$\varepsilon_1(n) = E[u_1^2(n)]$ is the mean squared error.

$$\frac{\partial \varepsilon_1(n)}{\partial \mathbf{w}_{21}(m)} = 0 \tag{2.29}$$

$2C_{u_1p_2}(m) = 0 \qquad m = 0,1,\dots,M-1$

The cross-correlation between the two signals $u_1(n)$ and $p_2(n)$is defined as [58]:

$$C_{u_1p_2}(m) = (h_{21}(m) - \mathbf{w}_{21}(m))\sigma_b^2 \tag{2.30}$$

$$\nabla_m = \frac{\partial C_{u_1p_2}(m)}{\partial \mathbf{w}_{21}(m)} = -\sigma_b^2 \tag{2.31}$$

$$\mathbf{w}_{21}^{(n)}(m) = w_{21}^{(n-1)} - \gamma_{21}\frac{C_{u_{jp_2}}(m)}{\nabla_m} \tag{2.32}$$

The idea of this algorithm is to replace the cross-correlation term with its instantaneous values and, with an appropriate choice ofγ_{21}, where $0 < \gamma_{21} < 2$, this implies that the algorithm remains stable$0 < \mu_{21} < 2/\sigma_2^2$.

We then obtain the following update formula:

$$\mathbf{w}_{21}^{(n)}(m) = \mathbf{w}_{21}^{(n-1)}(m) + \mu_{21}(u_1(n)p_2(n-m)) \tag{2.33}$$

Therefore, it can be stated that the SAD algorithm is equivalent to the LMS algorithm [45, 58].

- **Feed-forward SAD Algorithm (FSAD):**

Figure 2.14 shows the detailed structure of the FSAD algorithm.In this case, the update of the two adaptive filters is carried out using the cross-correlation vector between the two estimated output signals$u_1(n)\ and\ u_2(n)$.

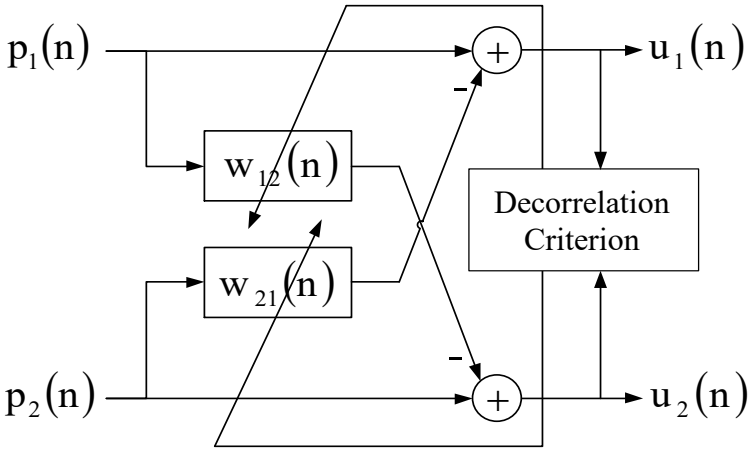

Figure 2.14. Structure of FSAD algorithm [58].

By adapting the update equation of the AD algorithm to the two-sensor forward structure [58], we obtain the following two update equations for the filters$\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$:

$$\mathbf{w}_{12}^{(n)}(m) = \mathbf{w}_{12}^{(n-1)}(m) + \mu_{12}(u_2(n)u_1(n-m))m = 0,1,\dots,M-1 \quad (2.34)$$

$$\mathbf{w}_{21}^{(n)}(m) = \mathbf{w}_{21}^{(n-1)}(m) + \mu_{21}(u_1(n)u_2(n-m))m = 0,1,\dots,M-1 \quad (2.35)$$

If the cross-correlation values between the two estimated signals are equal to zero, the update equations of the adaptive filters converge toward the true impulse responses. In other words, the two signals become decorrelated.

$$C_{u_1u_2}(m) = E[u_1(n)u_2(n-m)] = 0 \qquad \forall m \quad (2.36)$$

$$C_{u_2u_1}(m) = E[u_2(n)u_1(n-m)] = 0 \qquad \forall m \quad (2.37)$$

- **Feed-back SAD Algorithm (BSAD)**

In the backward structure shown in Figure 2.15, the decorrelation is performed between the two estimated output signals $v_1(n)$ and $v_2(n)$.

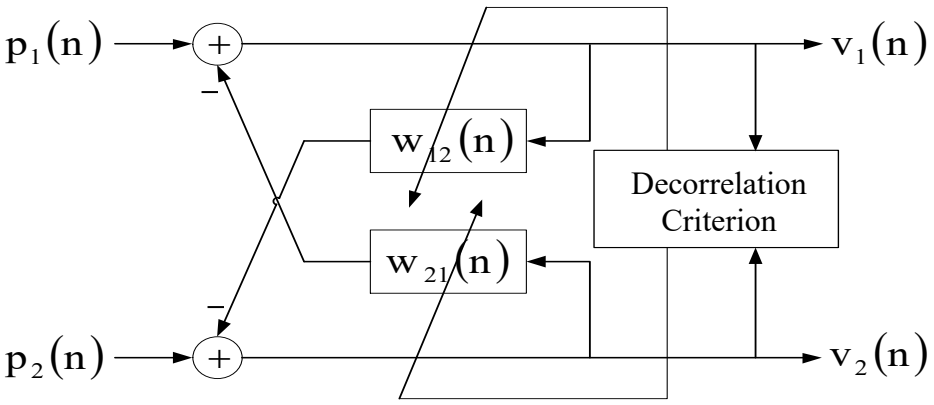

Figure 2.15. Structure of BSAD algorithm [58].

In this algorithm (BSAD), the two update equations of the adaptive filters$\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$ are given as follows:

$$\mathbf{w}_{12}^{(n)}(m) = \mathbf{w}_{12}^{(n-1)}(m) + \mu_{12}(v_2(n)v_1(n-m))m = 0,1,\dots,M-1 \quad (2.38)$$

$$\mathbf{w}_{21}^{(n)}(m) = \mathbf{w}_{21}^{(n-1)}(m) + \mu_{21}(v_1(n)v_2(n-m))m = 0,1,\dots,M-1 \quad (2.39)$$

Algorithm 2.3.SAD Algorithms [58].

FSAD	BSAD
$for\ n = 0,1,2,3, \ldots.$	$for\ n = 0,1,2,3, \ldots.$
Estimation of the output signals	*Estimation of the output signals:*
$u_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)p_2(n)$	$v_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)v_2(n)$
$u_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)p_1(n)$	$v_2(n) = p_1(n) - \mathbf{w}_{12}^T(n-1)v_1(n)$
Filter update equations	*Filter update equations*
$\mathbf{w}_{12}^{(n)}(m) = \mathbf{w}_{12}^{(n-1)}(m) + \mu_{12}(u_2(n)u_1(n-m))$	$\mathbf{w}_{12}^{(n)}(m) = \mathbf{w}_{12}^{(n-1)}(m) + \mu_{12}(v_2(n)v_1(n-m))$
$\mathbf{w}_{21}^{(n)}(m) = \mathbf{w}_{21}^{(n-1)}(m) + \mu_{21}(u_1(n)u_2(n-m))$	$\mathbf{w}_{21}^{(n)}(m) = \mathbf{w}_{21}^{(n-1)}(m) + \mu_{21}(v_1(n)v_2(n-m))$
End	*End*
Parameters and variables	
M: Adaptive filter length of $\mathbf{w}_{12}(n)$ and $\mathbf{w}_{21}(n)$ μ_{21}andμ_{12}:Fixed step size, m: Delay index, with$m = 0,1, \ldots, M-1$	

2.7. Conclusion

In this chapter, we provided a comprehensive overview of BSS and its applications to acoustic noise reduction, covering the main theoretical aspects, mathematical models, and key adaptive algorithms (2LMS, 2NLMS, and SAD) for two-sensor systems. However, these classical approaches suffer from a fundamental limitation: the trade-off between convergence speed and steady-state error imposed by a fixed step-size parameter.

To address this shortcoming, the next chapter proposes the **NN-V-FNLMS algorithm**, which integrates a neural network-based mechanism to dynamically estimate the optimal step-size, enabling both faster convergence and superior noise reduction performance.

Chapter III

New Two-Sensor Neural Networks Forward Algorithm for Acoustic Noise Reduction

Chapter 3: New Two-Sensor Neural Networks Forward Algorithm for Acoustic Noise Reduction

3.1. Introduction

In this chapter, we propose a novel Neural Network-based Variable-step-size (NN-V) estimation mechanism integrated into an efficient two-sensor acoustic noise reduction technique. Specifically, we introduce the NN-V-FNLMS algorithm, which overcomes the inherent convergence-error trade-off in the conventional FNLMS algorithm.

A simple single-layer Neural Network is trained on multiple types of acoustic noise signals to dynamically learn the relationship between important two-sensor signal characteristics and optimal step-size values. This adaptive NN-V mechanism enables the algorithm to dynamically adjust the step-size parameter, resulting in both faster convergence and superior speech enhancement in dispersive impulse response situations.

3.2. Two-channel convolutive mixing system

Acoustic signals in real-world settings (e.g., hands-free telephony, conference systems) are captured as convolutive mixtures rather than direct signals due to reverberation and propagation effects [61, 62]. This chapter focuses on the two-channel convolutive mixing system, which assumes two sources (speech and noise) and two sensors (microphones) (see Figure 3.1). As depicted in Figure 3.1.a, the speech signal (s(n)) and the noise signal (n(n)) each propagate to the microphones via paths characterized by a unique impulse response [63, 64]. This response mathematically models the complex effects of room acoustics, distance, and reflections on the signal.

The two noisy $ms_1(n)$and$ms_2(n)$, captured at the microphones, can be mathematically modeled as:

$$ms_1(n) = sp(n) * h_{11}(n) + n_1(n) + ns(n) * h_{21}(n) \quad (3.1)$$

$$ms_2(n) = ns(n) * h_{22}(n) + n_2(n) + sp(n) * h_{12}(n) \quad (3.2)$$

In practical acoustic signal processing scenarios, especially for real-time noise reduction applications, it is often beneficial to adopt a simplified model of the two-channel convolutive mixing system presented in Figure 3.1.b [65, 66].

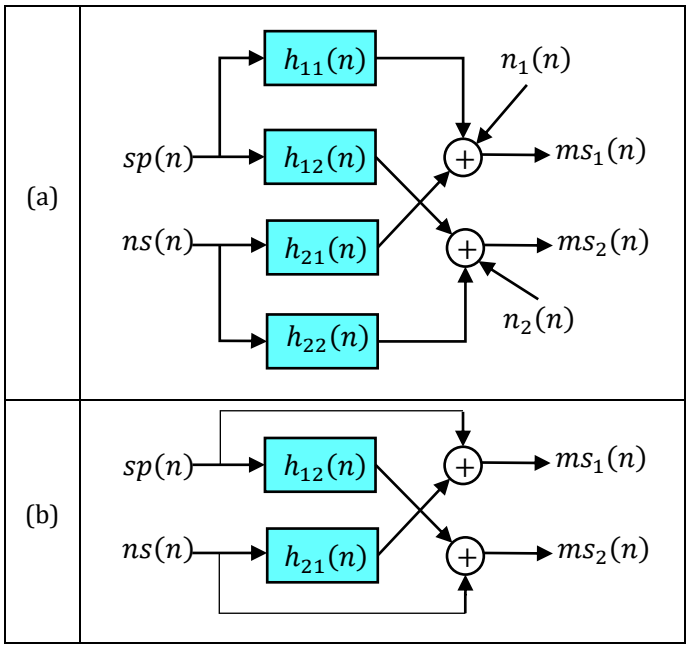

Figure 3.1. Two-channel acoustical convolutive system, (a) Full model and (b) Simplified model[64, 66].

The cross-path filters $h_{12}(n)$ and $h_{21}(n)$ still model the propagation of signals from the remote source to the non-adjacent microphone and are kept general [63, 66, 68]. Using these simplifications, the observed signals at microphones become:

$$ms_1(n) = sp(n) + ns(n) * h_{21}(n) \tag{3.3}$$

$$ms_2(n) = ns(n) + sp(n) * h_{12}(n) \tag{3.4}$$

3.3. Simplified two-sensor feed-forward NLMSalgorithm

This section details the formulation of Simplified Two-sensor Feed-forwardNLMSstructureadapted by NLMS algorithm as presented Figure 3.2.

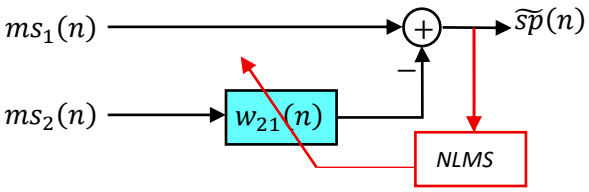

Figure 3.2. Simplified Two-sensor Feed-forward NLMS algorithm.

The enhanced speech $\widetilde{sp}(n)$is obtained by deducting the first mixing signal $ms_1(n)$from the output of the second adaptive filter, as given by Equation (3.5).

$$\widetilde{sp}(n) = ms_1(n) - ms_2(n) * w_{21}(n) \tag{3.5}$$

For the filter updates, we employ the fundamental NLMS algorithm. The update formula of the adaptive filter $\mathbf{w}_{21}(n)$ is provided in the following equation.

$$\mathbf{w}_{21}(n+1) = \mathbf{w}_{21}(n) + \mu_{21}\left[\frac{\widetilde{sp}(n)\mathbf{ms_2}(n)}{\varepsilon_{NLMS} + \|\mathbf{ms_2}(n)\|^2}\right] \quad (3.6)$$

We note that μ_{21} is the fixed step-size, and ε_{NLMS} is a small positive constant. The conditions$0 < \mu_{21} < 2$is required and sufficient, to guarantee the algorithm's convergence and stability in theMSE sense.

3.4.Proposed NN-V-FNLMS Algorithm

This section introduces the novel Neural Networks-based Variable Step-Size Feed-forward NLMS (NN-V-FNLMS) algorithm. We begin by detailing the two-sensor feed-forward structure and the optimal solutions for the adaptive filters, then present the time-domain formulations for the proposed algorithm.

We utilize the standard two-sensor Feed-forward, as depicted in Figure 3.2 but controlled by new variable step-size as presented in Figure 3.3. This structure aims to estimate the original source signals ($\widetilde{sp}(n)$) from the mixture signals ($ms_1(n)$and $ms_2(n)$) using one adaptive filter, $\mathbf{w}_{21}(n)$. The proposed NN-V-FNLMS algorithm adapts the classic FNLMS update rule by replacing the fixed step-size parameter with a dynamically predicted value derived from a Neural Network (NN).

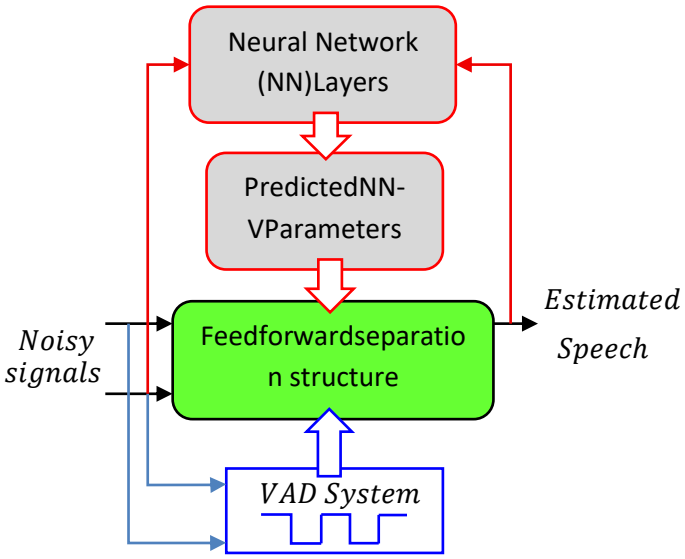

Figure 3.3. Global diagram of proposed algorithm.

The global model of the proposed two-sensor Feed-forward algorithm is presented in Figure 3.3. This structure is a hybrid framework consisting of three main components working together to achieve robust acoustic noise reduction:

Neural Network (NN) Layers: This module analyzes the input signals to predict the optimal step-size parameters. The NN Layers process feedback signals from the adaptive core and input signals to generate the Predicted NN-V Parameters (the variable step-sizes).

Voice Activity Detector (VAD) System: The VAD ensures that the adaptive filter updates are gated or supervised, allowing the algorithm to focus adaptation efforts primarily during periods of noise, or to switch between step-size policies based on the presence of speech. The output of the VAD can influence the NN training or directly gate the adaptive core's update.

Feed-forward Separation Structure (Adaptive Core): This is the fundamental two-sensor FF-BSS structure, responsible for processing the Noisy signals (the convolutive mixtures) to produce the Estimated Speech and estimated noise. It utilizes one adaptive filter $\mathbf{w}_{21}(n)$.

The adaptive filters within the Feed-forward separation structure are continuously guided by the Predicted NN-V Parameters determined by the Neural Network, replacing the fixed step-size of the conventional FNLMS.

3.4.1. Step-Size Estimation using Neural Networks (NN)

The core innovation is the NN-based Variable Step-Size ($\mu_{NN}(n)$) mechanism. A simple single-layer Neural Network is trained to map specific input signals (feedback and instantaneous power characteristics from the two-sensor system, potentially including VAD information) to an optimal step-size value at each time instant n. This dynamic step-size aims to increase its value during error periods (for fast convergence) and decrease it near convergence (for low steady-state error). The step-size $\mu_{NN}(n)$ is calculated as:

$$\mu_{NN}(n) = f_{NN}\{\mathbf{z}(n)\} \tag{3.7}$$

Where $\mathbf{z}(n)$ is the vector ofinput signals (signals flowing into the NN Layers block in Figure 3.4) and $f_{NN}\{\mathbf{z}(n)\}$is the function realized by the trained Neural Network.

The Neural Network is a Feedforward Network with a single hidden layer, often referred to as a Multilayer Perceptron (MLP). It is configured as follows:

Input Layer:

- Number of Inputs (Features): Two vectors
- Feature Vector (z): [p_{12};u_{12}] (squared values of two vectors signals, p1 and u1). These correspond to the instantaneous power characteristics used to guide the step-size.

Hidden Layer:

- Number of Neurons: 10
- Activation Function: A hyperbolic tangent sigmoid is used by default in MATLAB's standard new function for regression problems.

Output Layer:

- Number of Outputs (Labels): 1
- Output Value: The adaptive step-size, which is the variable being predicted.
- Activation Function: Linear for regression problems, which predicting a step-size.

The function new (features', labels', 10) sets up a two-layer network (one hidden layer) where the first argument (features') defines the input size, the second (labels') defines the output size, and the third (10) defines the size of the hidden layer.

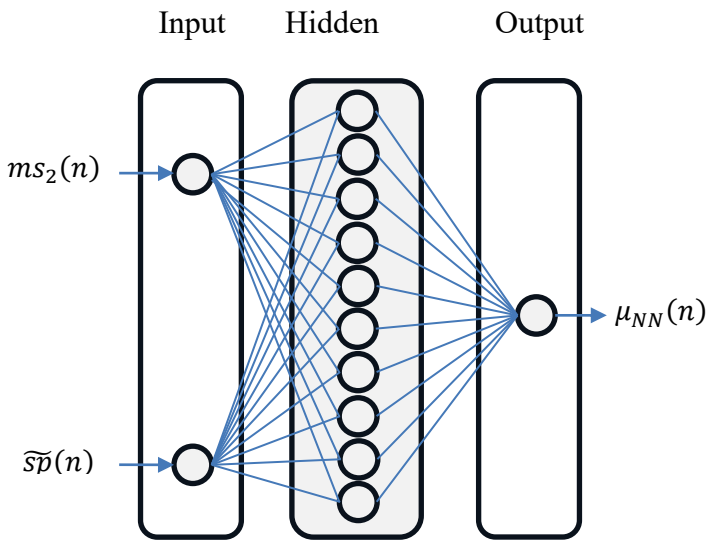

Figure 3.4. Detailed Neural Network (NN) Layers

3.4.2. Voice Activity Detector (VAD) System

The proposed NN-V-FNLMS algorithm incorporates a Voice Activity Detection (VAD) mechanism to guide its adaptive filtering process. VAD systems are typically used to

distinguish between segments containing speech and those containing only noise or silence, as illustrated in Figure 3.5.

Figure 3.5. Example of speech signal segmentation

This VAD information is then utilized to control how and when the filter coefficients are updated. In the NN-V approach, the adaptive filter $\mathbf{w}_{21}(n)$ processes the incoming noisy signal. A key feature is that $\mathbf{w}_{21}(n)$ is updated only during noise-only intervals (or inactive speech periods).

This selective adaptation strategy reduces the computational load and improves overall efficiency. Figure 3.6 shows a schematic of the VAD system, detailing how it manages the filter updates within the two-sensor Feed-forward configuration of the NN-V algorithm.

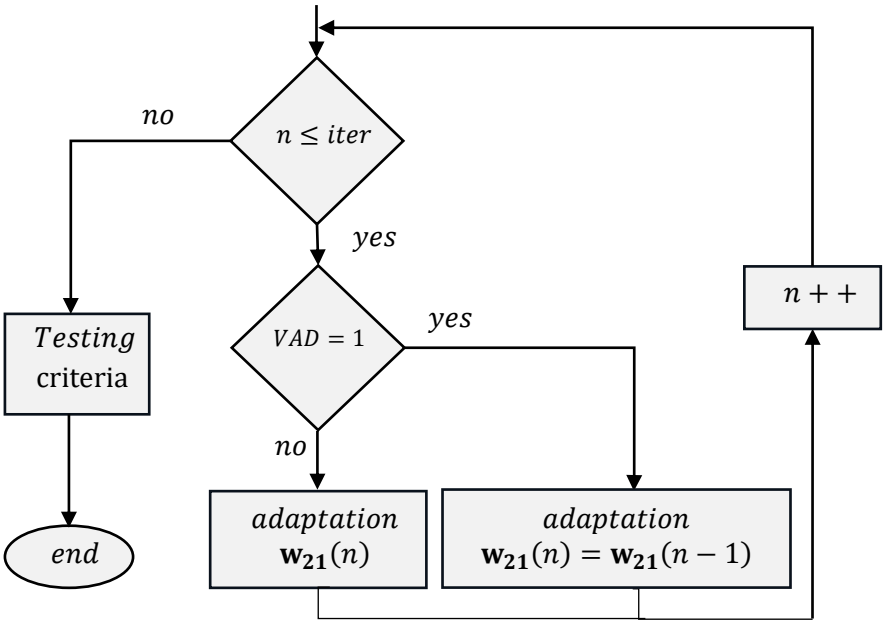

Figure 3.6. Voice activity detector (VAD) used for controlling the adaptation

3.4.3. Adaptive forward Separation Structure

To update the adaptive filters within the two-sensor forward structure (Figure 3.6), we propose using the NN-V-FNLMS algorithm. This approach modifies the standard NLMS

update rule by replacing the fixed step-size with a dynamic parameter estimated by the Neural Network (NN), and further controlling the adaptation using a Voice Activity Detector (VAD).

The filter coefficients for the adaptive filter $\mathrm{w}_{21}(\mathrm{n})$ are updated according to the following VAD-gated formula:

$$\mathbf{w}_{21}(n) = \begin{cases} \mathbf{w}_{21}(n-1) + \mu_{NN}(n)\dfrac{\widetilde{sp}(n)\boldsymbol{ms}_2(n)}{\varepsilon + \|\boldsymbol{ms}_2(n)\|^2} & if\ V = 0 \\ \mathbf{w}_{21}(n-1) & if\ V = 1 \end{cases} \tag{3.8}$$

$\mu_{NN}(n)$represents the adaptive step-size estimated by Neural Network Layer, with the necessary and sufficient condition to guarantee the convergence and the stability of this algorithm in the MSE sense is $0 < \mu_{NN}(n) < 2$, and ε presents a very small positive constant.The tap-weight vectors of the adaptive filters $\mathbf{w}_{21}(n)$ is defined respectively by,

$\mathbf{w}_{21}(n) = \left[w_{21,1}(n), w_{21,2}(n), \dots, w_{21,M}(n)\right]^T$.

We also define vector (last M values) of inputs noisy signal, $ms_2(n)$ as:

$\boldsymbol{ms}_2(n) = [ms_2(n), ms_2(n-1), \dots, ms_2(n-M+1)]^T$

where M is the length of adaptive filter.

V is the VAD status (typically 0 or 1), which gates the adaptation. This control ensures the filter primarily updates during noise-only or inactive speech intervals, as determined by the VAD system (Figure 3.6).

3.5. Simulations and Results

3.5.1. Signals and parameters

This section details a simulation analysis to evaluate the DL-VSS-FNLMS algorithm in various noisy acoustic settings. The experiments used a realistic convolutive mixing model, combining two sources: Source 1 (Speech presented in Figure 3.7), a 4-second, phonetically balanced French sentence from the AURORA database (8 kHz,16-bit) [65, 66]**,** and Source 2 (Noise presented in Figure 3.8), consisting of real-world disturbances like white, babble, and aircraft noise.

The final mixture signal was created by convolving each source with a distinct room impulse response (see Figure 3.9), resulting in challenging noisy observations (presented in Figure 3.10) with an input Signal-to-Noise Ratio (SNR) of −6 dB using model presented in [67]. This setup allows for a robust and fair assessment, aiming to demonstrate the proposed DL-VSS method's benefits in improvingboth convergence speed and speech quality.

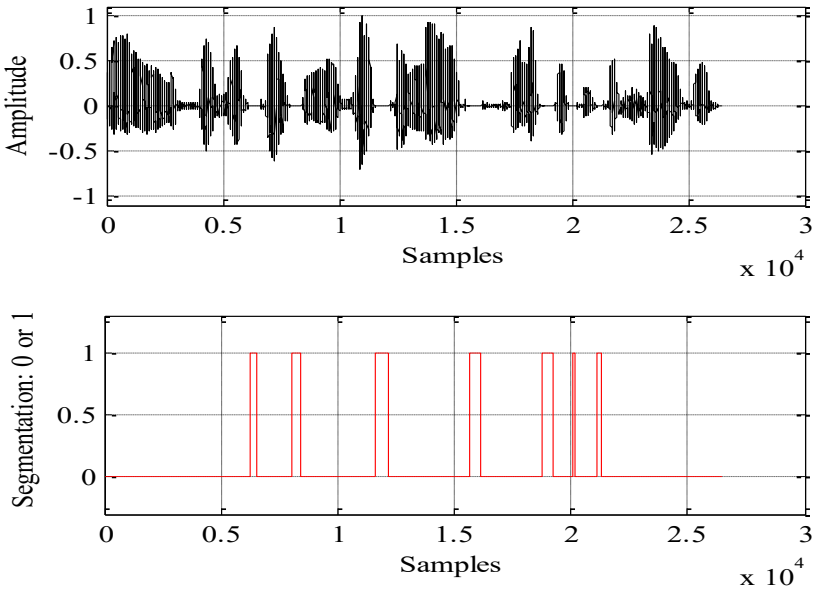

Figure 3.7. Original speech signal and generated segmentation (VAD)

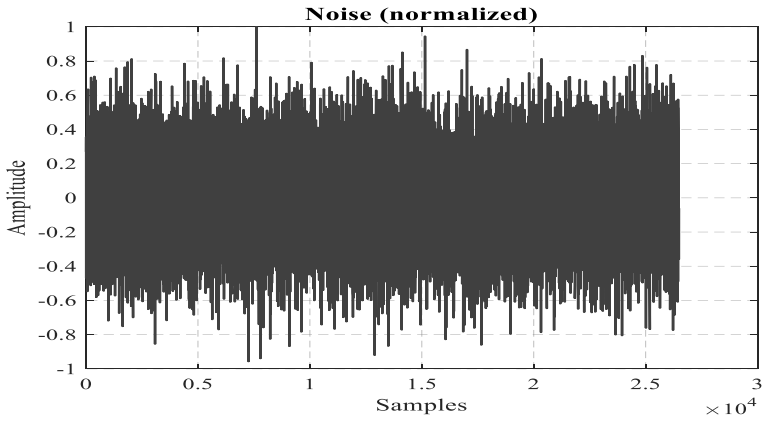

Figure 3.8. Acoustic noise signal

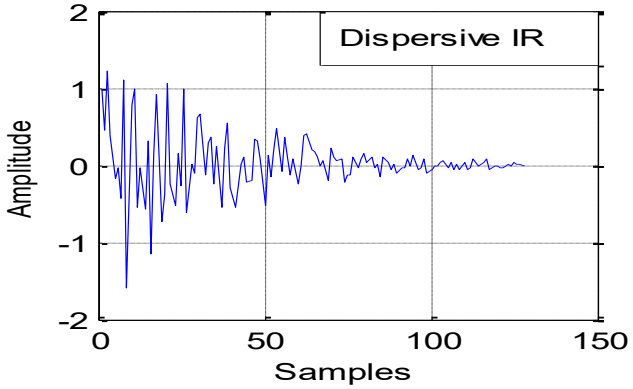

Figure 3.9. Examples of real dispersive impulse response h_{21}

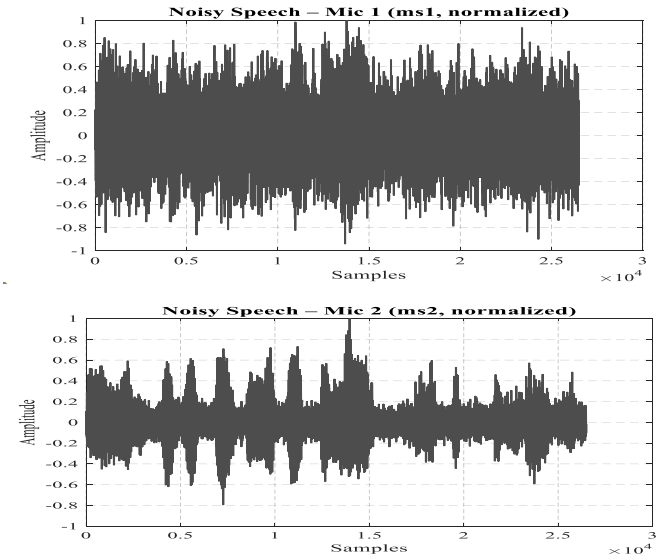

Figure 3.10. Two noisy speech signals with Input-SNR = - 6 dB

The evaluation of the proposed advanced methods used a consistent set of input signalsand fixed/adaptive parameters to ensure a fair comparison across various acoustic conditions, modeling acoustic propagation with length dispersive impulse responses is M=128. Internal algorithm parameters, like the maximum adaptive step sizes and constants (λ=0.67, β=2, and ρ=10^{-6}), were carefully tuned on a validation set.

3.5.2. Time evolution of VSS and enhanced speech

InFigures3.11 and 3.12, we present respectively the variation of the step-size parameter and evolution of enhanced speech signal.

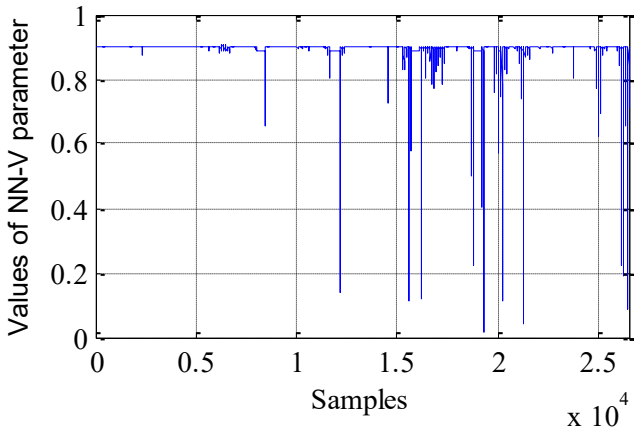

Figure 3.11.NN-V step-size variation

The Figure 3.11 presents the NN-V Step-Size Variation, illustrates the core adaptive mechanism of the Proposed NN-V-FNLMS algorithm. This figure shows the calculated values of the variable step-size parameter, $\mu_{NN}(n)$,which are dynamically predicted by the

trained Neural Network (NN). The step-size exhibits rapid and significant fluctuations, frequently soaring close to its maximum value (near 1) and plunging towards zero. These high values are strategically generated during periods where the adaptive filter needs to converge quickly, such as at the start of the process or when a sudden change in the acoustic noise requires a fast update of the filter weights. Conversely, the swift drops to low values occur when the filter is approaching its optimal solution or when the Voice Activity Detector (VAD) detects the presence of speech, minimizing the filter's updates to maintain a low steady-state error and prevent the algorithm from introducing distortion into the desired speech signal. This continuous, data-driven adjustment by the NN demonstrates how the proposed algorithm effectively resolves the inherent convergence-error trade-off by adapting its learning rate based on instantaneous signal characteristics.

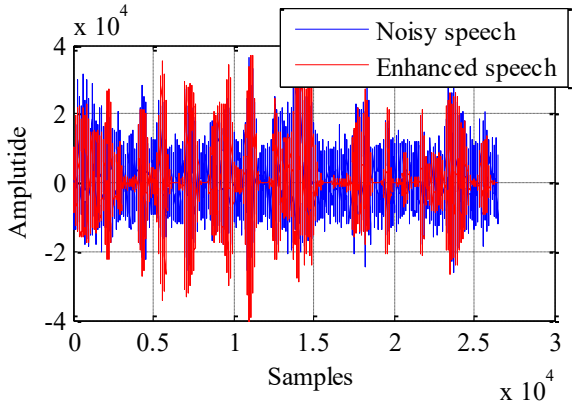

Figure 3.12. Time evolution of enhanced speech and noisy ones.

The Figure 3.12, visually confirms the noise reduction capability of the proposed NN-V-FNLMS algorithm. The plot contrasts the Noisy speech signal, which represents the initial, highly corrupted audio captured at the microphone with a challenging input Signal-to-Noise Ratio (SNR) of -6 dB, against the resulting Enhanced speech signal. The noisy signal exhibits large and erratic amplitude fluctuations, characteristic of significant background noise interference.

The enhanced speech signal, while tracking the overall envelope of the speech, displays markedly lower amplitude and reduced high-frequency oscillations during periods that appear to be non-speech segments. This noticeable decrease in the baseline signal amplitude directly translates to a successful suppression of the background acoustic noise by the adaptive filter. This time-domain comparison provides clear evidence that the NN-V-FNLMS algorithm is effective in separating the desired speech from the unwanted noise, delivering a cleaner, higher-quality output signal.

3.5.3.MSE Evaluation

The segmental mean square error (SegMSE) is employed to evaluate the noise-reduction performance of each sparse algorithm. Its value at the system output is computed using the next expression:

$$(SegMSE_{\lambda})_{dB} = 20\ log_{10}\left(\sum_{i=0}^{B-1}|v_1(i)|^2\ VAD_{\lambda}\right) \tag{3.9}$$

In this formulation, denotes the estimated speech signal, while B represents the time-averaging frame length of the output signal (with $B = 128$ used in all SegMSE simulations). The term corresponds to a voice activity detector (VAD), which identifies periods containing only noise. Since the SegMSE metric is assessed exclusively during speech-absent intervals, it is calculated solely when the signal consists of acoustic noise components.

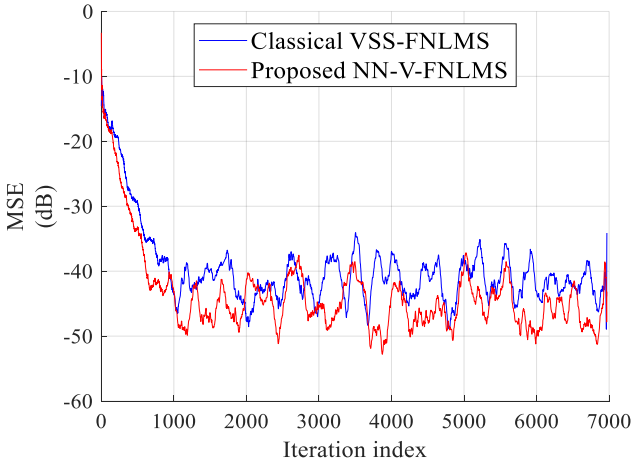

Figure 3.13.MSE evaluation obtained bay classical and proposed algorithms.

The Figure 3.13 quantitatively demonstrates the superior convergence characteristics of the Proposed NN-V-FNLMS algorithm compared to the Classical VSS-FNLMS. MSE, measured in dB, reflects the power of the residual error, making it a critical criterion for assessing the algorithm's ability to estimate the target signal. The plot reveals that the NN-V-FNLMS achieves a faster initial convergence rate, evidenced by the steeper, more rapid decline in its MSE curve during the initial phase (Blocs 0 to 1000). This quick descent is directly attributed to the Neural Network's mechanism, which intelligently predicts a large step-size when the error is high, thus accelerating the learning process. Crucially, in the steady-state regime after convergence, the proposed algorithm maintains a lower average steady-state MSE (around -50 to -55 dB), showcasing its ability to achieve a smaller residual error than the classical method. This dual advantage of faster convergence and lower steady-state error confirms that the NN-

V approach successfully overcomes the inherent trade-off present in conventional fixed and variable step-size algorithms.

3.5.4. System Mismatch (SM)

The System Mismatch (SM)indicator quantifies the average deviation between the true acoustic impulse response coefficients (h_{21}) and the coefficients produced by the adaptive filter (w). This metric provides essential insight into the convergence behavior and numerical stability of the adaptive filtering algorithm. Lower SM values correspond to better system identification accuracy and improved algorithmic convergence toward the optimal filter solution.

$$[SM_n]_{dB} = 20\ log_{10}\left[\frac{\|\mathbf{h}_{21} - \mathbf{w}_{21}(n)\|}{\|\mathbf{h}_{21}\|}\right] \tag{3.10}$$

The System Mismatch is evaluated over data segments of 128 samples., allowing for localized evaluation of convergence behavior and temporal variations in the adaptive filtering process.

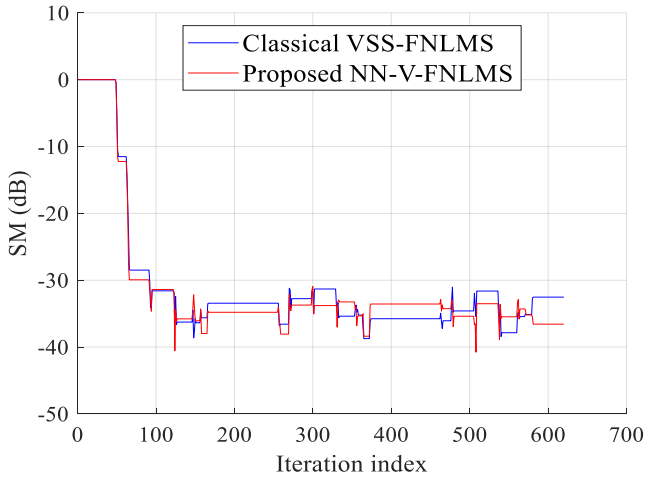

Figure 3.14.SM evaluation obtained bay classical and proposed algorithms.

The Figure 3.14provides insight into the convergence quality and stability of the adaptive filter coefficients by measuring how far they are from the optimal solution. Comparing the Proposed NN-V-FNLMS algorithm against the Classical VSS-FNLMS, the proposed method demonstrates a superior ability to stabilize the filter.

In the initial phase, the NN-V-FNLMS achieves a faster reduction in SM, meaning its filter weights converge more rapidly to the ideal values due to the dynamically adjusted step-size.

More importantly, the proposed algorithm maintains a lower average steady-state SM (around -40 to -45 dB) after convergence compared to the classical method. Since System Mismatch directly correlates with the quality of the filter's estimation, this lower steady-state value confirms that the Neural Network-based variable step-size mechanism leads to a more accurate and stable adaptive filter, resulting in better overall system performance.

3.5.5. Segmental SNR (Seg-SNR)

The Segmental SNR evaluates the degree of signal quality enhancement introduced by the noise reduction process on a frame-by-frame basis, with a particular focus on the degree of noise suppression across short-time segments of the speech signal. It provides a more detailed assessment of performance than global SNR by reflecting local variations in speech and noise energy over time.

$$[SegSNR_n]_{dB} = 10\ log_{10}\left[\frac{\sum_{i=1}^{N}|sp(i)|^2}{\sum_{i=1}^{N}|sp(i)-\widetilde{sp}(i)|^2}\right]\ if\ \ VAD(i)\ = 1 \qquad (3.11)$$

It is computed as the signal-to-noise ratio over short, fixed-length frames of 512 samples, allowing a localized assessment of enhancement performance. Higher Seg-SNR values reflect more effective noise suppression and improved preservation of speech content.

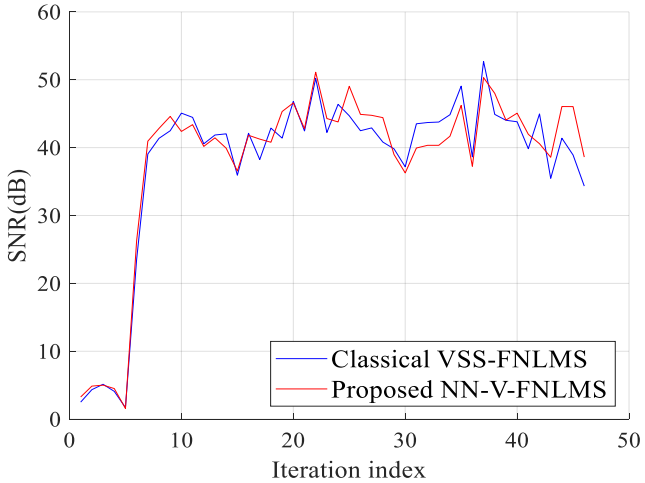

Figure 3.15. SegSNREvaluation obtained bay classical and proposed algorithms.

The Figure 3.15, illustrating the Output Segmental Signal-to-Noise Ratio (SegSNR) Evaluation, serves as the ultimate measure of enhanced speech quality, as SegSNR is widely accepted for its strong correlation with human perception of noise reduction. Comparing the proposed NN-V-FNLMS algorithm with the Classical VSS-FNLMS, the proposed method

demonstrates its superior effectiveness in improving speech clarity. Across virtually the entire duration of the evaluation (Blocs 10 to 45), indicating that the NN-V-FNLMS consistently achieves a higher Output SegSNR. Since a higher SegSNR signifies greater noise suppression relative to the speech signal, this result confirms that the dynamic step-size selection, guided by the Neural Network, and the VAD-gated adaptation collectively lead to a more robust and perceptually superior noise reduction than the classical method. The sustained higher SegSNR, with peaks reaching around 50 to 55 dB, proves that the proposed algorithm successfully enhances the quality of the noisy speech by effectively suppressing the background acoustic noise.

3.6. Conclusion

The Proposed NN-V-FNLMS algorithm marks a significant advancement in acoustic noise reduction by successfully overcoming the classic convergence-error trade-off through the integration of a Neural Network for dynamic step-size prediction. Simulation results conclusively demonstrate its superiority over the Classical VSS-FNLMS: the intelligent NN control provides both faster initial convergence and a lower steady-state error in the MSE and SM evaluations. Most importantly, the algorithm achieves a consistently higher Output SegSNR, confirming that the dynamic adaptation, governed by the NN and gated by the VAD, leads to a perceptually higher-quality enhanced speech signal. This innovative approach validates the use of a simple neural network to learn complex acoustic characteristics, establishing a robust and highly effective framework for real-world two-sensor noise suppression.

Chapter IV

New Advanced Adaptive Feed-Forward Algorithm based on Variable Step-Size Deep Learning Estimation

4.1.Introduction

In this chapter, we introduce a novel two-sensor feed-forward NLMS adaptive noise reduction system enhanced by a deep-learning-based variable step-size estimation framework. The core innovation lies in training a Deep Neural Network (DNN) to intelligently estimate the optimal step-size by learning the complex relationship between power-based signal features and desired step-size values under various noisy conditions.

The proposed strategy employs an MSD supervised learning mechanism and utilizes a VAD assisted adaptation policy that exclusively processes noisy frames. This dynamic, data-driven modulation of the learning rate provides faster convergence, achieves superior residual noise suppression, and ultimately yields improved speech quality, particularly in sparse and highly reverberant acoustic environments, demonstrating a clear advantage over classical VSS-based approaches.

4.2. Structure of proposed DL-VSS two-sensor adaptive feed-forward algorithm

In this chapter, we present a two-sensor feed-forward noise-suppression framework incorporating a deep-learning-driven variable step-size strategy. The core idea of the proposed approach is to improve the convergence characteristics and noise-attenuation capability of traditional adaptive filtering schemes by continuously adjusting the step-size in accordance with the acoustic conditions.

The architecture relies on a classical feed-forward structure coupled with a deep neural model responsible for estimating, in real-time, the most suitable step-size for the adaptive filters. The development process begins by establishing the theoretical basis of the optimal step-size for adaptive filtering systems. Subsequently, we derive the time-domain expression of the suggested DL-VSS-FNLMS algorithm and describe its integration within a dual-microphone configuration.

An overview of the overall system design is depicted in Figure 4.1, where the neural network supervises the step-size adaptation and the adaptive filters process the noisy signals captured by the two sensors.

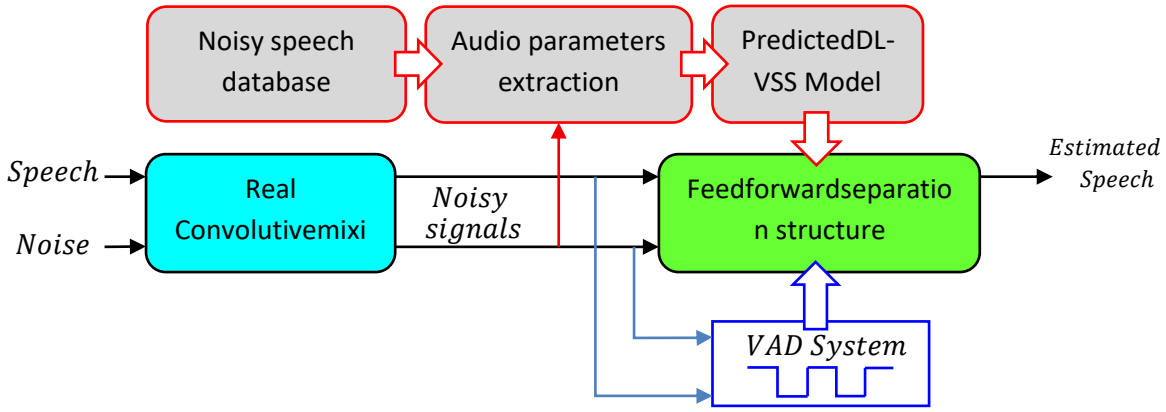

Figure 4.1. Global structure of proposed DL-VSS-FNLMS

It is important to emphasize that conventional dual-sensor feed-forward systems are founded on the hypothesis that the reference and primary signals are statistically independent. In other words, the desired speech component and the noise component are assumed to be uncorrelated, which can be expressed mathematically as $E[sp(n)\, ns(n-m)] = 0, \forall m$

This assumption guarantees that the two input sources do not share common statistical dependencies.The proposed DL-VSS-FNLMS scheme extends this classical framework by incorporating a deep-learning-guided variable step-size mechanism for adaptive noise filtering. In this methodology, a curated noise dataset is merged with clean speech samples, and the resulting mixtures are produced through a convolutive acoustic model to emulate realistic acoustic environments.

A deep neural network is subsequently trained to infer, in real time, the appropriate adaptation step-size for the adaptive filters, enabling robust performance under diverse noise conditions.By learning the relationship between the acoustic context and the appropriate step-size values, the model dynamically guides the NLMS adaptation process, thereby enhancing convergence speed and robustness under diverse noise conditions.

The proposed algorithm incorporates a VAD module to distinguish between speech-active and noise-only regions. This mechanism governs the adaptive update process by enabling step-size adjustment only when speech is present, while freezing or limiting adaptation during silence or noise-only segments. Such selective updating prevents unnecessary filter adaptation, minimizes the influence of noise-dominant frames, and significantly improves stability and noise suppression performance. In addition, this strategy reduces computational

overhead during inactive speech intervals, leading to a more efficient and context-aware learning process. The detailed architecture of the proposed DL-VSS-FNLMS system is illustrated in Figure 4.2.

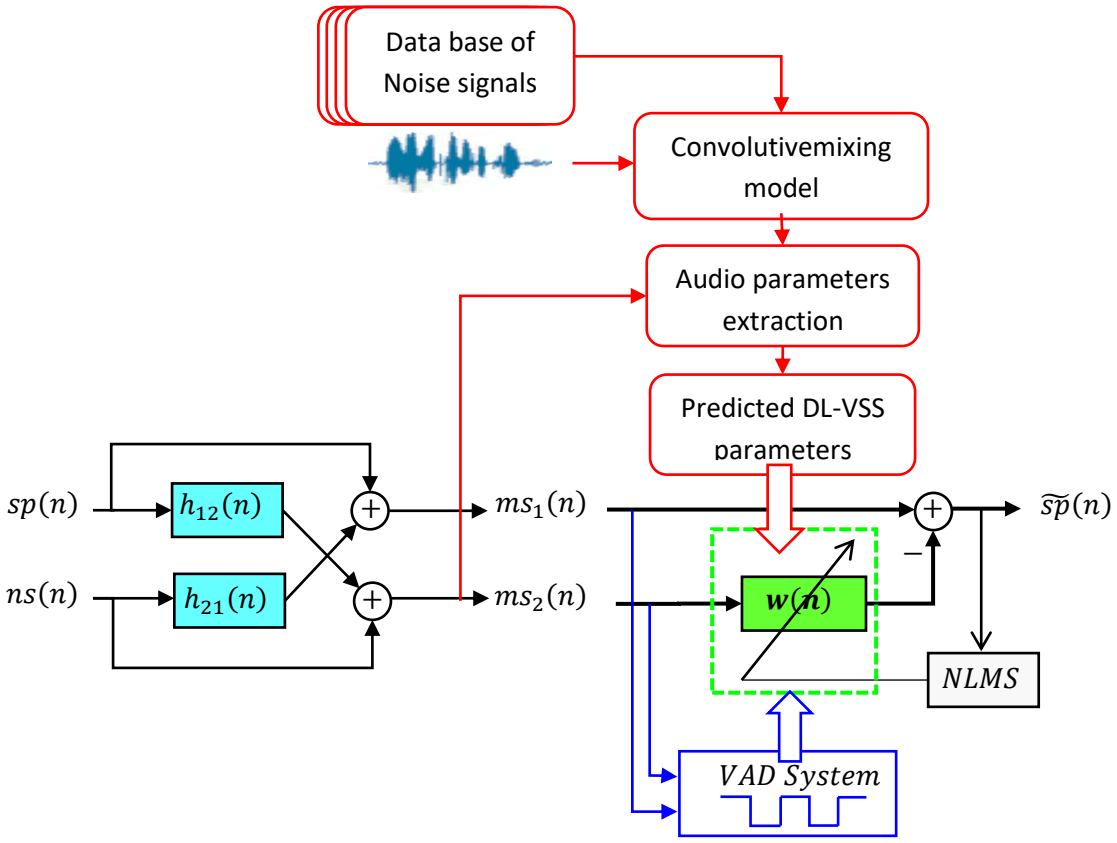

Figure 4.2. Architecture of the proposed DL-VSS-FNLMS algorithm.

4.3. VADmechanism

In the proposed DL-VSS-FNLMS framework, a voice activity detection (VAD) module is incorporated to supervise and adjust the adaptive filtering procedure. The function of the VAD is to distinguish frames that contain speech from those dominated by noise-only segments or silence. By identifying speech-active intervals, as depicted in Figure 4.3, the system can selectively control the step-size adaptation and filtering behavior. This information provides a reliable decision cue that ensures the adaptive filter updates are performed only when relevant speech information is present, thereby improving robustness and convergence efficiency.

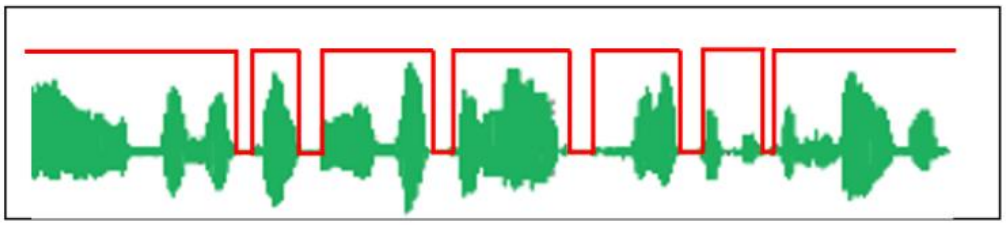

Figure 4.3. Example of speech signal segmentation using voice activity detector (VAD)

By selectively enabling the adaptation only during speech-present intervals and freezing or limiting updates during noise-only frames, the algorithm avoids erroneous coefficient adjustments caused by noise fluctuations. Consequently, the VAD enhances the robustness of the adaptation process, reduces computational waste, and contributes to improved speech enhancement performance, particularly in non-stationary acoustic environments.

In the proposed DL-VSS framework, the adaptive filter w(n) processes the noisy microphone input to enhance the speech signal. A central aspect of this strategy lies in the selective adaptation of the filter: w(n) is updated only during noise-only segments or periods where speech activity is absent. This VAD-controlled update mechanism prevents unintended filter modifications during speech, thereby avoiding speech distortion while improving convergence reliability.

By restricting updates to noise-dominant intervals, the computational burden is reduced and the learning process becomes more efficient and stable. Figure 4.4 illustrates the VAD-driven adaptation control scheme within the two-sensor feed-forward DL-VSS architecture, highlighting how the detector regulates coefficient updates based on the detected speech state.

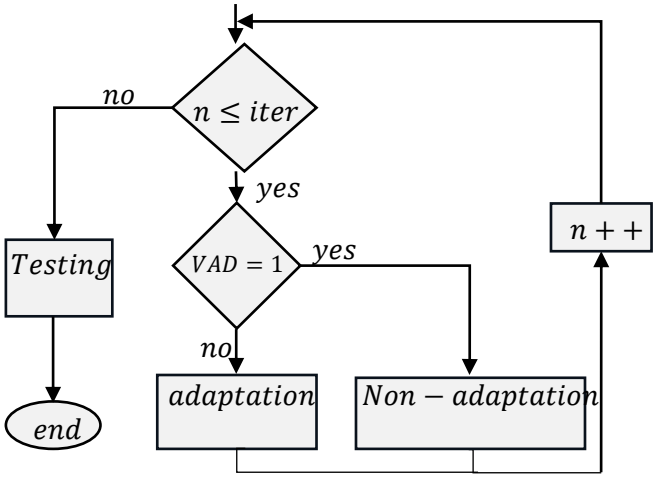

Figure 4.4. The control of adaptive filer by VAD system

To improve the flexibility and robustness of the dual-sensor feed-forward scheme, the adaptation step-size $\mu_{DL}(n)$ is automatically regulated through a deep-learning model. By continuously adjusting this parameter, the algorithm maintains fast convergence while ensuring stable tracking of the adaptive filter coefficients $w(n)$.

The neural network responsible for generating the step-size is trained using feature representations extracted from noisy speech signals. Only segments identified as noise-dominant by the VAD module are used during the learning phase, allowing the model to capture the behavior of the adaptive filter under challenging acoustic conditions.

To strengthen the adaptability and stability of the proposed dual-sensor feed-forward framework, the step-size coefficient $\mu_{DL}(n)$ is continuously adjusted using a deep-learning-driven estimation module. By modulating this parameter in real time, the system preserves a balance between rapid convergence and precise tracking of non-stationary acoustic conditions, thereby enhancing the performance of the adaptive filter $w(n)$.

In the proposed framework, the variable step-size is generated by a neural network trained to infer optimal learning rates from relevant signal features extracted during noise-only intervals, as determined by the VAD module. By relying on speech-free frames, the model avoids bias introduced by speech components and learns a reliable mapping between noise characteristics and appropriate step-size values. This data-driven adaptation strategy enables more accurate and context-aware control of the filter dynamics, leading to improved noise reduction performance under non-stationary acoustic conditions.

4.4. Adaptive filter formulation of proposed DL-VSS-FNLMS algorithm

To update the adaptive filter$w(n)$, a two-sensor feed-forward structure is adopted and combined with the NLMS algorithm. In the proposed method, however, the adaptation process is governed by the deep-learning-based variable step-size parameters instead of a fixed or analytically derived step-size. This learning-based control enhances the flexibility and robustness of the adaptive process under non-stationary acoustic conditions.
The filter update rule is selectively applied according to the decision of the VAD module. That is, the adaptive filter coefficients are adjusted only when noise-only frames are detected, ensuring reliable learning and preventing distortion during active speech segments. The

resulting update equation of the proposed DL-VSS-FNLMS algorithm is expressed as follows:

$$\boldsymbol{w}(n) = \begin{cases} \boldsymbol{w}(n-1) + \mu_{DL}(n)\dfrac{\widetilde{sp}(n)\boldsymbol{ms}_2(n)}{\varepsilon + \|\boldsymbol{ms}_2(n)\|^2} & if\ VAD = 0 \\ \boldsymbol{w}(n-1) & if\ VAD = 1 \end{cases} \quad (4.1)$$

The term $\mu_{DL}(n)$represents the adaptive step-size estimated by the deep learning model. To ensure convergence and stability of the proposed algorithm in the mean-square error (MSE) sense, this parameter must satisfy the classical constraint

$0 < \mu_{DL}(n) < 2$, The constant ϵ denotes a very small positive value introduced to avoid numerical instability and division by zero during the adaptation process.

The tap-weight vector of the adaptive filter $w(n)$ is defined as follows:

$$\boldsymbol{w}(n) = [w_1(n), w_2(n), \dots, w_M(n)]^T.$$

We also define the input vector, consisting of the last M samples of the noisy signal $ms_2(n)$, as follows:

$$\boldsymbol{ms}_2(n) = [ms_2(n), ms_2(n-1), \dots, ms_2(n-M+1)]^T$$

Where*M*represents the length (number of taps) of the adaptive filter.

4.5. Proposed variable step-size minimization strategy

To guarantee both rapid convergence and reliable estimation of the optimal filter weights, the step-size is obtained through an optimization procedure that minimizes the mean-square deviation (MSD) between the adaptive filter coefficients and the ideal impulse response. In this context, the goal is to identify the optimal deep-learning-based step-size $\mu_{DL}^{opt}(n)$ for the adaptive filter $w(n)$, such that the discrepancy between the estimated response and the ideal system response h_{21} is minimized.

This formulation guarantees that the learned step-size not only accelerates convergence but also preserves long-term stability by forcing the adaptive filter to track the true acoustic transfer function with minimal steady-state error.

$$\boldsymbol{\xi}(n) = \boldsymbol{h}_{21} - \boldsymbol{w}(n) \quad (4.2)$$

To evaluate the tracking performance of the adaptive filter, the MSD metric is employed. The MSD measures the discrepancy between the adaptive filter coefficients $w(n)$ and the optimal impulse responseh_{21}. It isdefined as follows:

$$MSD(n) = E[\|\boldsymbol{\xi}(n)\|^2] \tag{4.3}$$

To characterize the adaptive behavior of the proposed algorithm, we derive the evolution of the MSD as a function of the step-size parameter. Starting from the MSD definition and applying the squared Euclidean expansion together with standard assumptions on the input signal statistics (independence assumption and narrowband approximation), the recursive MSD expression can be expressed as follows:

$$MSD(n) - MSD(n-1) = \mu_{DL}^2 E\left[\frac{\left(\widetilde{sp}(n)\right)^2}{\sigma_2(n)}\right] - 2\mu_{DL} E\left[\frac{\boldsymbol{\xi}^T(n-1)\widetilde{\mathbf{ms}}_{\mathbf{2}}(n)\widetilde{sp}(n)}{\sigma_2(n)}\right] \tag{4.4}$$

To guarantee a progressive reduction in the estimation error, the MSD must satisfy

$MSD(n) - MSD(n-1) < 0$, under the condition that$\mu_{DL}^{opt}(n) < 2\,\nabla(n)$.

Here, $\nabla(n)$ denotes a small term associated with the cross-correlation between the input and output of the adaptive filter. Instead of explicitly computing statistical expectations, a recursive approximation is employed, offering a computationally efficient and numerically stable solution.

$\mu_{DL}^{opt}(n) = \mu_{max}\widetilde{\nabla}(n)$, with the estimated quantities $\widetilde{\nabla}(n)$ is given by,

$$\widetilde{\nabla}(n) = \mu_{max}\left[\frac{\|\mathbf{Q}(n)\|^2}{\rho + \|\mathbf{Q}(d)\|^2}\right] \tag{4.5}$$

Where$\mathbf{Q}(n)$captures the cross-correlation dynamics between the input and output signals of the adaptive filter. It is progressively updated over time using the following recursive estimation mechanism:

Where$\mathbf{Q}(n)$ represents the evolving cross-correlation term that reflects the interaction between the adaptive filter's input and output signals. This quantity is iteratively updated through the following recursive estimation procedure:

$$\mathbf{Q}(n) = \lambda\, \mathbf{Q}(n-1) + \frac{(1-\lambda)}{\sigma_2(n) + \varepsilon}\, \widetilde{sp}(n)\widetilde{\mathbf{ms}}_2(n) \tag{4.6}$$

With$0 < \lambda_i < 1$. This theoretical optimal value $\mu_{DL}^{opt}(n)$is subsequently employed as the supervisory training target for a deep neural network. The network is trained to infer the optimal step-size adaptively from a set of discriminative features extracted during noise-only intervals. In doing so, the model learns to emulate the minimization behavior of the MSD criterion, while eliminating the need for explicit MSD computation during real-time operation. Thus, during inference, the step-size is generated in a purely data-driven manner, ensuring fast and stable adaptation without additional analytical cost.

4.6. Proposed deep learning–based estimation of variable step-size parameters

To ensure reliable estimation of the adaptive step-size $\mu_{DL}(n)$ in the proposed dual-sensor feed-forward NLMS system, the deep neural network must be supplied with features that accurately represent both the temporal evolution and spectral content of the noisy microphone signal $ms_2(n)$. The objective is to ensure that the model can reliably infer the optimal adaptation rate under varying acoustic environments and noise conditions.

This section describes the methodology adopted for feature construction and learning. Specifically, the deep learning-based VSS estimation pipeline is structured around three major stages:

*(i) Noisy Speech Dataset Construction:*A database combining clean speech signals and diverse real-world noise types is generated using the previously defined convolutive mixing model to emulate realistic degraded acoustic scenarios.

*(ii) Acoustic Feature Extraction:*Discriminative time-domain and energy-based descriptors are computed from the noisy microphone signal during noise-only frames (as detected by the VAD module). These features capture the local signal power, residual energy, and statistical variations that influence optimal learning dynamics.

*(iii)Deep Learning-Driven Step-Size Prediction:*The extracted features are provided to a neural network trained to approximate the optimal step-size values based on the MSD-driven minimization strategy introduced earlier. Once trained, the model predicts $\mu_{DL}(n)$ online, thus enabling real-time, data-driven adaptation without explicit analytical MSD computation.

The complete deep learning-assisted VSS estimation architecture is illustrated in Figure4.5, high the interaction between the noisy signal input, feature extraction block, neural network predictor, and adaptive filtering stage.

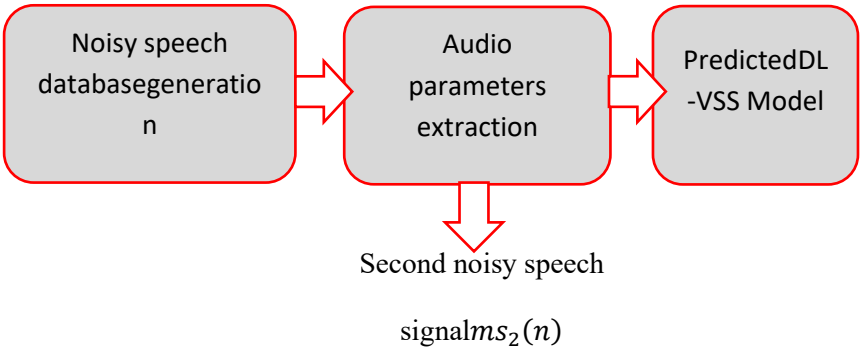

Figure 4.5 Three parts of deep learning VSS parameters extraction

4.6.1. Construction of the noisy speech Database

In this study, a realistic noisy-speech corpus was generated using a convolutive mixing strategy that emulates real acoustic propagation scenarios. The simulation setup accounts for the interaction between clean speech and the acoustic environment by incorporating two representative classes of room impulse responses: dispersive and sparse models [71-73].

Dispersive impulse responses exhibit long-tail, densely distributed coefficients and describe acoustic settings where sound undergoes multiple reflections and reverberation, such as large reverberant halls or wide indoor spaces. Conversely, sparse impulse responses are characterized by a small number of dominant reflections, representative of acoustically controlled or low-reverberation environments, such as compact meeting rooms or treated studios. By modeling both acoustic conditions, the constructed dataset provides a challenging and diverse training scenario, enabling the neural model to adaptively learn robust step-size behavior across highly dynamic acoustic contexts.

For training and evaluation, clean speech recordings were extracted from the TIMIT speech corpus [74], which offers high-fidelity, multi-speaker utterances recorded under well-controlled conditions. These clean speech signals constitute the target reference to be reconstructed by the adaptive filtering system.

Environmental noise signals were obtained from the NOISEX-92 database [75], a widely used benchmark for speech enhancement and noise reduction research. Multiple noise types were selected to cover both stationary and highly non-stationary acoustic conditions, including white noise, babble speech noise, F-16 cockpit noise, factory noise, high-frequency channel noise, and buccaneer noise. The noise components were then convolved with the clean speech using a dual-microphone acoustic mixing model. Simulating realistic multi-path acoustic propagation scenarios typical of hands-free communication and embedded microphone systems.

The resulting dataset consists of a diverse collection of noisy speech mixtures spanning different noise intensities, temporal fluctuations, and spectral properties (Figure 4.6). This variability is essential for enabling the neural network to learn step-size patterns that remain reliable in highly dynamic and unpredictable acoustic conditions.

This database forms the foundation of the training pipeline for the proposed DL-VSS adaptive filtering framework, promoting strong generalization and reliability in real-world deployments.

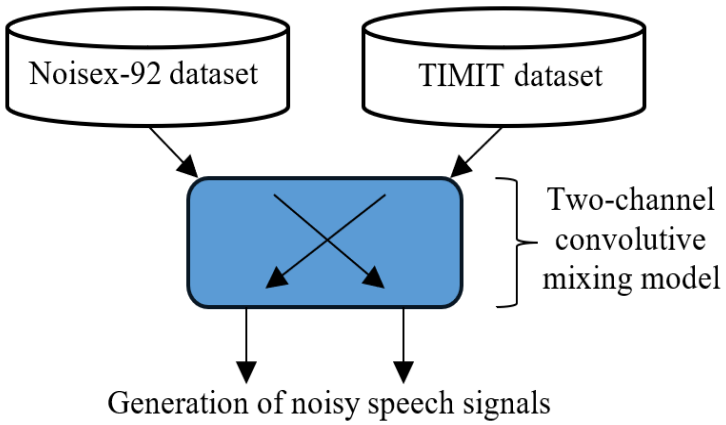

Figure 4.6.Noisy Speech Database Generation

4.6.2. Audio Feature Extraction

The feature extraction process begins with signal normalization to reduce amplitude inconsistencies and ensure stable learning dynamics. Following normalization, a set of perceptually motivated and cepstral representations are computed to capture both the spectral content and temporal dynamics of the noisy speech signal.

The MFCCs are first extracted, given their widespread use in speech processing and strong ability to represent phonetic information and vocal tract characteristics. To enhance spectral resolution and capture fine auditory cues, GTCCs are also computed. Both feature types are extended by incorporating their first and second temporal derivatives (Δ and Δ^2), which provide sensitivity to rapid spectral variations typically observed in speech signals subjected to noisy environments.

In addition, energy-based auditory features derived from the ERB, Bark, and Mel frequency scales are calculated using a dedicated audio feature extraction framework. These perceptual descriptors model the auditory system's nonlinear frequency resolution and contribute complementary information regarding signal energy distribution across critical bands.

All extracted coefficients are merged into a single feature vector that characterizes each analysis frame, thereby forming a high-dimensional and discriminative representation of the input signal. To reinforce the model's robustness to speech activity variations, a silence removal / speech activity filtering stage is applied based on frame-level energy thresholds. This ensures that only informative frames particularly those corresponding to background noise segments relevant for step-size estimation are retained for model training.

This extensive feature representation delivers a detailed characterization of the acoustic environment, allowing the deep learning model to capture the temporal–spectral patterns needed for precise step-size estimation under time-varying noise conditions.

(i) Second Noisy Speech Signal Pre-Processing:

The second noisy speech signal$ms_2(n)$obtained from the two-sensor feed-forward configuration undergoes a preprocessing stage prior to feature extraction and step-size estimation. This stage aims to improve data stability and ensure consistent dynamic range for subsequent deep learning processing.

To avoid numerical instability and amplitude saturation effects, the signal is first normalized so that its values lie within the interval [−1,1]. This operation reduces the influence of amplitude variations and facilitates a more robust learning process.

The normalizationisperformedaccording to:

$$ms_{2,n}(n) = \frac{\mathbf{ms}_2(n)}{max|\mathbf{ms}_2(n)|} \tag{4.7}$$

The main purpose of normalizing the noisy speech signal is to improve numerical stability across the processing pipeline by keeping the signal amplitude within a controlled and consistent range.

This prevents numerical overflow, precision loss, or scaling artifacts during feature extraction and adaptive learning stages. Additionally, normalization ensures consistency and comparability among extracted acoustic features by reducing sensitivity to variations in recording gain, microphone distance, or speaker loudness. Consequently, the deep learning model becomes more robust and capable of generalizing across heterogeneous speech inputs, independent of their original amplitude levels or acquisition conditions.

(ii) MFCC Parameter Design:

The MFCCs provide a compact and perceptually aligned description of the signal's short-term spectrum and are extensively used in speech analysis tasks. In this work, MFCCs are integrated into the feature set to supply the neural network with relevant spectral information needed for accurate step-size prediction.

Since speech signals are inherently non-stationary, they must be processed in short time intervals during which their statistical properties can be approximated as quasi-stationary. Therefore, before MFCC computation, the signal is segmented into overlapping frames based on the following configuration:

- Window length:

A frame duration of 25 ms is selected, computed as:$L_W = \text{round}(0.025 \times f_s)$,
Where*fs* is the sampling frequency. This choice offers an effective compromise between temporal precision for transient phonetic events and frequency resolution for spectral analysis.

- Hop length (frame shift):

Consecutive frames are shifted by 10 ms, $L_H = \text{round}(0.01 \times \text{f}_s)$. This overlap ensures smooth temporal continuity, preserves fine acoustic transitions, and minimizes information loss between adjacent frames.

This frame-based processing ensures that MFCCs accurately capture both the articulatory and spectral dynamics of the noisy speech signal while providing a stable and informative representation for input into the proposed deep learning-assisted adaptive filtering model.

$$L_{overlap} = L_W - L_H \tag{4.8}$$

To reduce spectral leakage resulting from abrupt discontinuities at the frame boundaries, each analysis frame is first multiplied by a Hamming window before computing its spectral representation. The Hamming window provides a smooth tapering of the signal toward the frame edges, thereby reducing high-frequency distortions in the frequency domain. This progressive attenuation enhances the reliability of the short-time spectral representation and contributes to more accurate and robust MFCC estimation, particularly in the presence of noise.

The Hamming window is defined as:

$$Hw(n) = 0.54 - 0.46 \times cos\left(\frac{2\pi n}{L_W - 1}\right), \qquad 0 \leq n < L_W \tag{4.9}$$

(iii) MFCC Coefficient Extraction:

After windowing and preprocessing, 13 MFCCs are computed for each frame. The MFCC extraction pipeline consists of four major signal-processing stages designed to emulate the human auditory system and produce compact spectral representations:

- Short-Time Fourier Transform (STFT)

Following windowing, each frame is processed using the STFT to obtain a frequency-domain representation, from which the magnitude spectrum is computed.

- Mel-Filterbank Processing

The magnitude spectrum is passed through a bank of triangular filters distributed according to the Mel-scale, which reflects the nonlinear frequency resolution of the human auditory system.

- Log-Energy Compression

A logarithmic non-linearity is applied to approximate perceptual loudness scaling and compress dynamic range.

- Discrete Cosine Transform (DCT)

The log-filterbank energies are decorrelated using the DCT, producing a compact set of cepstral coefficients. Only the first 13 coefficients are retained, as they contain the most perceptually relevant information.

This process results in a set of MFCC features that concisely capture the short-term vocal tract characteristics and spectral envelope of the speech signal, making them suitable for robust deep learning-based modeling in noisy environments.

The mathematical formulation is given by:

$$MFCC(t) = DCT\left(log\left(MelSpectrum\left(mx_{2,n}(t)\right)\right)\right) \quad (4.10)$$

Where $mx_{2,n}(t)$denotes the time-domain speech frame centered at time index n. The function MelSpectrum(·) applies a triangular Mel-scale filterbank to the magnitude spectrum of the signal, while the logarithmic operation models the nonlinear loudness sensitivity of the human auditory system. Finally, the Discrete Cosine Transform (DCT)decorrelates the log-Mel coefficients and compacts the spectral information into a low-dimensional cepstral representation.

The Mel-filterbank energy output for the m-th filter is expressed as:

$$E_m = log\left(\sum_k \left|MX_{2,n}[k]\right|^2 \times H_m[k]\right) \quad (4.11)$$

where $MX_{2,n}[k]$ denotes the DFT of the windowed noisy frame $mx_{2,n}(n)$, $H_m[k]$represents the m-th Mel filter, and $\left|MX_{2,n}[k]\right|^2$ corresponds to the power spectral density of the noisy signal in the frequency domain. A logarithmic compression is then applied to approximate the nonlinear sensitivity of the human auditory system.

After Mel filtering and log-compression, the cepstralcoefficients$c_n$for the n-th frame are obtained by applying the DCT as follows:

$$c_n = \sum_{m=1}^{M} E_m \times cos\left(n \times \frac{\pi}{M} \times \left(m - \frac{1}{2}\right)\right), 0 \leq n < N_{coeffs} \quad (4.12)$$

M is the number of Mel filters, N_{coeffs} is the number of desired cepstral coefficients (typically 12 or 13), and It converts the *M_logMel* energies into N_{coeffs}decorrelatedcepstral coefficients.

In this formulation, M corresponds to the total number of Mel filters and N_{coeffs} denotes the number of cepstral coefficients extracted (typically 12 or 13). The transformation maps the *M_logMel*energies into N_{coeffs}uncorrelated cepstral coefficients.

(iv) Extended Spectral Feature Extraction:

In addition to standard MFCC features, a broader set of perceptually inspired spectral descriptors was extracted to enrich the acoustic representation. The goal of this stage is to capture complementary spectral information that strengthens the robustness and generalization ability of the deep learning model, especially in noisy and reverberant environments. These complementary features encode various psychoacoustic aspects of the human auditory system.

- *ERB Spectrum:*

The ERB framework reproduces the cochlear frequency-selectivity mechanism, mimicking the auditory filter shapes observed in the human hearing system, where auditory filters exhibit bandwidths that vary with frequency. Unlike uniformly spaced spectral features, ERB-based analysis provides greater resolution at lower frequencies and coarser resolution at higher frequencies, aligning closely with auditory perception.

Given the short-time Fourier magnitude spectrum $\left|MX_{2,n}(f)\right|^2$ of the noisy frame, the ERB spectrum for frame n is computed by passing the power spectrum through a bank of ERB-spaced auditory filters $H_m^{ERB}(f)$:

$$ERB_m(n) = log\left(\sum_f \left|MX_{2,n}(f)\right|^2 \times H_m^{ERB}(f)\right) \quad (4.13)$$

where $MX_{2,n}(f)$ denotes the Fourier transform of the noisy speech frame centered at time index n, and $H_m^{ERB}(f)$ represents the frequency response of the m-thGammatone filter in the ERB-scale filterbank, with m=1,2,…,M and M being the total number of ERB bands.

where $MX_{2,n}(f)$ refers to the Fourier transform of the noisy speech frame centered at time index nnn, and $H_m^{ERB}(f)$ denotes the frequency response of the mmm-thGammatone filter in the ERB-scale filterbank, with m=1,2,…,M and M indicating the total number of ERB bands.

- *Bark Spectrum:*

The Bark spectrum provides another perceptually motivated spectral representation, based on the Bark scale, which divides the frequency axis into critical bands according to psychoacoustic models of auditory frequency grouping. This scale reflects how the human auditory system processes sound frequencies, assigning finer resolution to lower-frequency regions and broader integration at higher frequencies. As a result, the Bark spectrum is particularly effective in characterizing perceptually salient variations in the spectral envelope, making it a valuable complement to MFCC and ERB features, especially in noisy or reverberant acoustic environments.

- *Mel Spectrum:*

The Mel spectrum is derived from the Mel scale, a perceptual frequency mapping that approximates the human auditory system's sensitivity to pitch. This scale allocates linear resolution to low-frequency regions and logarithmic resolution to high-frequency regions, thereby providing finer spectral detail where most speech energy and formant information are concentrated. Such emphasis on lower frequencies makes the Mel spectrum particularly effective for speech analysis and enhancement tasks.

- **Cepstral Features: GTCC and Derivatives**

GTCCs constitute a perceptually motivated cepstral representation that parallels MFCCs, but replaces the Mel filterbank with a Gammatone auditory filterbank, which more closely models the frequency selectivity characteristics of the human cochlea. Unlike triangular Mel filters, Gammatone filters emulate the nonlinear frequency tuning properties of the basilar membrane, offering enhanced discrimination in low-frequency regions and more realistic auditory processing.

GTCCs are particularly robust in noisy and reverberant environments, as the auditory-inspired filtering process preserves key speech cues while attenuating irrelevant noise components. This makes them a useful complementary feature to MFCCs for deep learning-based speech enhancement and acoustic modeling tasks.

Let$mx_{2,n}(n)$ denote the time-domain noisy speech frame centered at time index n. The GTCC computation begins by transforming this frame into the frequency domain using a windowed STFT:

$$MX_{2,n}(k) = SFTT\left(mx_{2,n}(n)\right) \tag{4.14}$$

Power Spectrum

$$P_t(k) = \left|MX_{2,n}(k)\right|^2 \tag{4.15}$$

A Gammatonefilterbank is then applied$G_m[k]$ (with$M$ filters):

$$E_m(t) = \sum_k P_t(k) \times G_m[k] \tag{4.16}$$

Logarithmic Compression

$$\tilde{E}_m(t) = log(E_m(t) + \varepsilon) \tag{4.17}$$

Withε is a small constant to avoid $log(0)$

Discrete Cosine Transform (DCT)

$$GTCC_n(t) = \sum_{m=1}^{M} \tilde{E}_m(t) \times cos\left(\frac{n\pi}{M} \times \left(m - \frac{1}{2}\right)\right), 0 \leq n < N_{coeffs} \tag{4.18}$$

Delta Coefficients (ΔGTCC) and Delta-Delta Coefficients (Δ²GTCC) correspond to the first- and second-order temporal derivatives of the GTCC features. They capture the dynamic evolution of the spectral characteristics over time, which is essential for accurately representing non-stationary signals such as speech.

The ΔGTCC coefficients reflect the first-order variation (rate of change) of the GTCC features:

$$\Delta GTCC_n(t) = \frac{\sum_{l=0}^{L} l \times \left(GTCC_n(t+l) - GTCC_n(t-l)\right)}{2 \times \sum_{l=0}^{L} l^2} \tag{4.19}$$

Here, L denotes the window length used for computing the temporal derivatives, which is typically set to 2

Δ²GTCC refers to the second-order temporal derivatives of the GTCC features. These coefficients describe the acceleration of spectral variations over time, providing valuable information about the dynamic evolution of the acoustic signal.

$$\Delta^2 GTCC_n(t) = \frac{\sum_{l=0}^{L} l \times \left(\Delta GTCC_n(t+l) - \Delta GTCC_n(t-l)\right)}{2 \times \sum_{l=0}^{L} l^2} \tag{4.20}$$

The extended spectral feature representation at time index ttt can be expressed as the concatenation of the static GTCC coefficients and their first- and second-order temporal derivatives.

$$AFE(t) = [GTCC_n(t), \Delta GTCC_n(t), \Delta^2 GTCC_n(t), Mel(t), Bark(t), EBR(t)] \tag{4.21}$$

This rich and multidimensional representation substantially increases the expressive capacity of the input features, enabling the neural network to more effectively model and adapt to complex and dynamic acoustic environments.

(v)Feature Fusion:

Following the extraction of spectral and cepstral descriptors, including MFCCs, multiple perceptually motivated spectral representations (Mel, Bark, and ERB), as well as GTCCs and their corresponding temporal derivatives, all components are concatenated to form a single feature matrix. This fusion step aggregates complementary speech characteristics, resulting in a richer and more informative acoustic representation for subsequent learning stages.

By integrating features obtained from diverse perceptual and spectral domains, the resulting matrix jointly captures both the static structure and temporal evolution of the speech signal. Such a multi-dimensional feature space supplies the deep learning model with a highly discriminative and robust input representation, enabling improved speech–noise separation and facilitating better adaptation to complex, non-stationary acoustic environments.

$$F_t = [MFCC(t), AFE(t)] \tag{4.22}$$

The chosen feature set is specifically tailored to support step-size adaptation rather than phonetic classification tasks. Frame energy contributes to the stabilization of $\mu_{DL}(n)$ during

transient signal bursts. MFCCs and their first-order derivatives (ΔMFCC) capture near-end speech leakage, which could otherwise lead to speech distortion if the adaptive step size becomes excessively large.

GTCCs and their temporal derivatives (ΔGTCC) offer greater robustness against noise and effectively track narrowband or colored interference. Meanwhile, ERB, Bark, and Mel representations provide smoother, low-variance spectral envelopes, which are particularly beneficial under very low signal-to-noise ratio (SNR) conditions.

(vi) Automaticsilencedetection:

To enhance the quality and balance of the training data used for step-size estimation, an automatic silence-detection mechanism was incorporated. The frame-level energy was computed as the squared norm of each feature vector, providing a reliable cue for distinguishing speech-active regions from silence.

$$E_t = \sum_{i=1}^{D} F_{t,i}^2 \tag{4.23}$$

Where$D$ denotes the dimensionality of the feature vector. A threshold corresponding to the 10^{th} percentile of the overall energy distribution was employed to identify silent frames within the dataset.

Here, D refers to the length of the feature vector. To isolate silent segments, an energy threshold equal to the 10^{th} percentile of the global distribution was applied, ensuring that only the lowest-energy frames were labeled as silence.

$$Silence\ Frames = \{t | E_t < percentile(E, 10)\} \tag{4.24}$$

These low-energy segments are particularly informative, as they typically represent periods during which the optimal step-size should be near zero. Including them in the training set enables the network to learn appropriate behavior in both speech-active and silent regions, enhancing overall stability and adaptation capability.

Figure 4.7 illustrates the complete sequence of steps involved in the audio feature extraction process, integrating all the subcomponents described in the preceding sections.

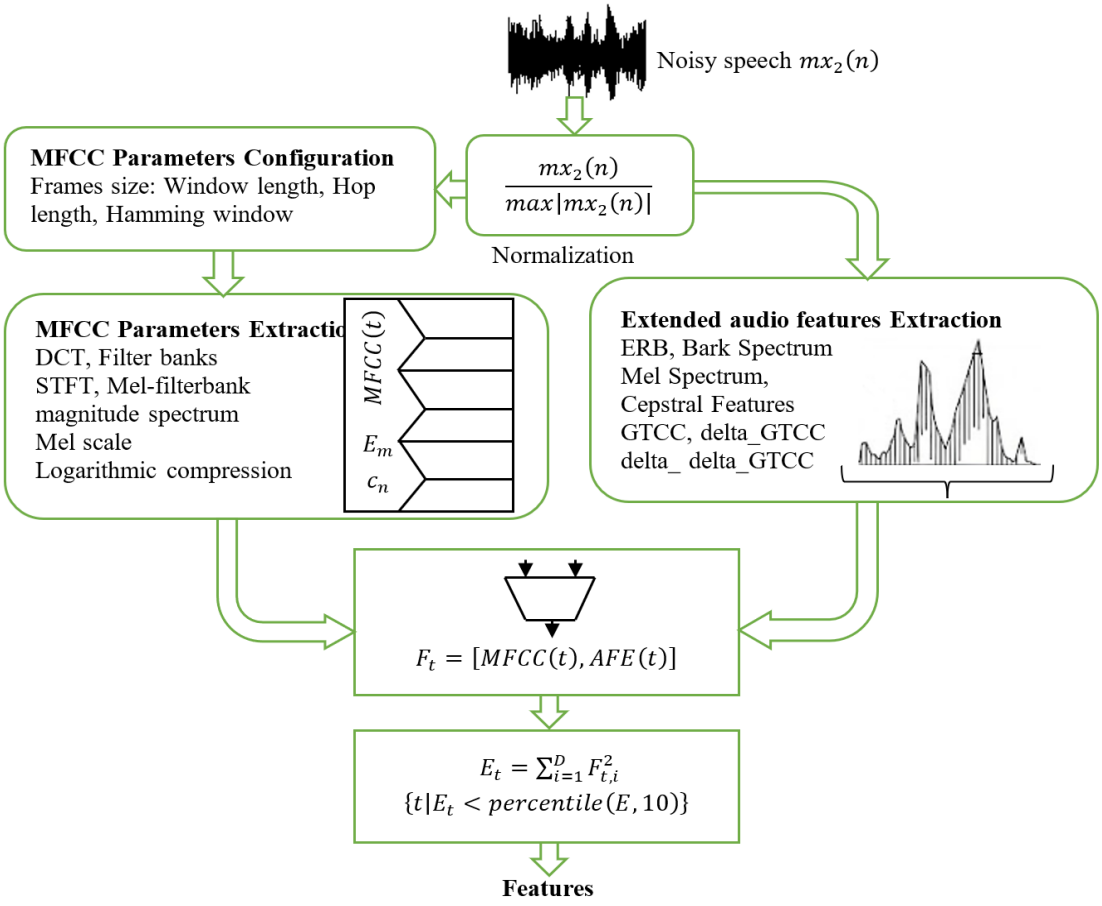

Figure 4.7. Audio Feature Design and Extraction Strategy for the Deep Learning Framework

4.7. Neural Network Framework for Adaptive Step-Size Prediction

In this study, a deep-learning-driven approach is introduced to estimate the adaptive step-size term $\mu_{DL}(n)$ within a dual-microphone NLMS structure. The objective is to infer this parameter directly from noise-contaminated speech features, allowing it to vary in real time according to the acoustic context. The proposed solution relies on a recurrent neural network with stacked LSTM layers, chosen for their ability to capture temporal structure and long-term dependencies inherent in audio sequences.

(i) Dataset Preprocessing and Normalization:

The feature inputsdescribed in subsection 4.6 serve as the basis for the proposed network. The model's input consists of a comprehensive feature matrix constructed from the previously extracted representations, including MFCCs, GTCCs, and additional spectral features such as

ERB, Bark, and Mel coefficients. These descriptors are concatenated to form a rich feature representation for each time frame of the noisy speech signal.

The ground-truth labels consist of the step-size values $\mu(n)$, obtained for every frame following the analytical procedure described in subsection 4.5. The resulting dataset is then randomly partitioned into training and testing sets using a 70/30 ratio to enable fair performance assessment and ensure good generalization capability.

To ensure numerical stability and consistent feature scaling during model training, z-score normalization is applied to all input features. The mean and standard deviation used for this operation are derived solely from the training set, preventing any leakage of information into the evaluation phase. This process ensures that all feature dimensions contribute equally during optimization and helps accelerate convergence. Finally, the normalized feature matrices are transposed and formatted as sequential time-series inputs suitable for processing by the RNN architecture.

(ii) Model Architecture:

The proposed RNN structure, depicted in Figure 4.8, is tailored to learn the temporal dynamics of the acoustic feature sequence and to generate the adaptive step-size values required by the dual-microphone NLMS system. The network consists of multiple stacked LSTM layers, chosen for their proven ability to capture long-term temporal relationships in sequential audio data.

The main architectural components are outlined as follows:

- Input Layer:

The network begins with a *sequence input layer* configured to receive time-sequential acoustic data. Its input dimensionality corresponds to the number of concatenated acoustic features per frame, ensuring compatibility with the feature representation described in subsection 4.6.

- First LSTM Block:

The first recurrent block consists of an LSTM layer with 128 memory units. It processes the temporal sequence and captures short- to medium-term dependencies across frames. A dropout layer with a rate of 20% follows this LSTM to mitigate overfitting and enhance model generalization.

- Second LSTM Block:

The second LSTM layer, comprising 64 units, is stacked atop the first to enable deeper temporal representation learning. This layer further refines the model's ability to capture longer-term dependencies. A dropout of 20% is again applied to stabilize training and reduce co-adaptation between neurons.

- Third LSTM Block:

The third and final recurrent block contains 32 LSTM units and operates in *output mode = 'last'*, which propagates only the final hidden state to subsequent layers. This configuration allows the network to encode the entire temporal sequence into a compact latent representation that retains the most salient information for the prediction task.

- Fully Connected Layers:

The condensed temporal representation is fed into two fully connected (dense) layers: (i) The first dense layer includes 64 neurons followed by a ReLU activation function, introducing non-linearity and enabling the modeling of complex mappings. (ii) The second dense layer contains a single neuron that outputs the predicted continuous-valued step-size coefficient for the corresponding time frame.

- Output Layer:

A regression layer serves as the final component, mapping the scalar output from the dense layer to the real-valued target$\mu_{DL}(n)$. This design enables the network to perform frame-level regression, aligning its predictions with the analytically derived step-size reference values.

(iii) Training Methodology:

To effectively train the proposed RNN for adaptive step-size prediction, a structured training procedure was adopted. This approach was formulated to promote stable convergence, enhance generalization, and ensure efficient use of computational resources. The key components of the training process are summarized below:

- Optimization Algorithm:

The network parameters were optimized using the Adam optimizer, a widely adopted stochastic gradient–based method that integrates the benefits of both the Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp). Adam facilitates efficient and stable convergence by employing adaptive learning rates and

momentum estimates for each model parameter, making it particularly suitable for non-stationary and high-dimensional data such as speech features.

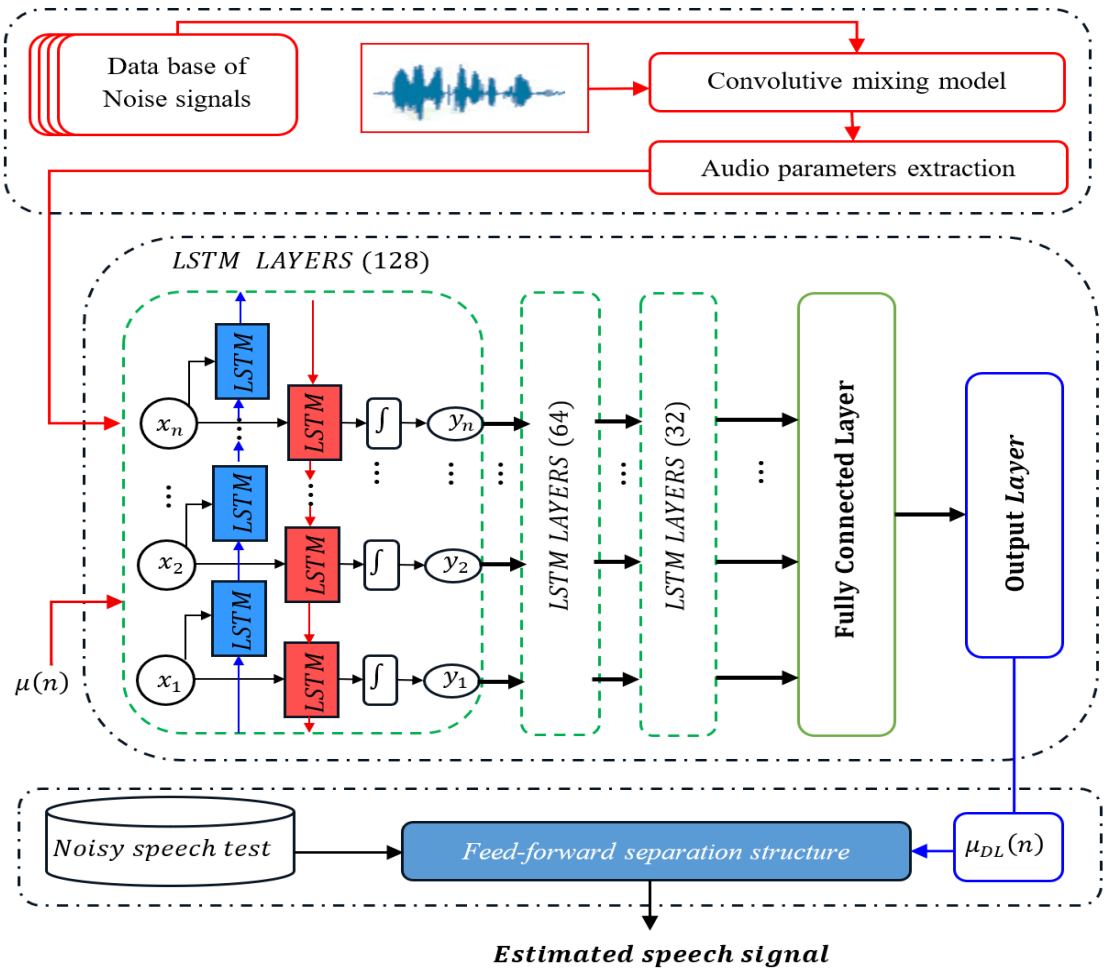

Figure 4.8. Detailed deep learning model used for acoustic noise reduction

- Training Time:

The model was trained for 100 full epochs, allowing the network adequate iterations to learn complex temporal patterns and reduce prediction errors. Empirical experimentation indicated that this duration offers a suitable compromise between performance gains and computational effort.

- Mini-Batch Setup:

A mini-batch size of 32 was used for training. This choice provides a balanced trade-off between stable parameter updates and computational efficiency, which is especially important for sequential data processing where memory usage can become substantial.

- Shuffling Policy:

To improve generalization and reduce overfitting caused by temporal ordering effects, the training samples were reshuffled after each epoch. This step breaks any artificial sequence patterns in the data, encouraging more robust and reliable learning behavior.

- Loss Tracking and Custom Callbacks:

A custom callback was incorporated into the training process to log and observe the training loss after every epoch. This mechanism enables continuous monitoring of learning behavior and supports early detection of issues such as stagnation, divergence, or overfitting.

- Software Environment:

The complete training workflow covering network construction, data preprocessing, loss tracking, and optimization was carried out using MATLAB's Deep Learning Toolbox. The *trainNetwork* function was utilized to perform iterative optimization and ensure a consistent and reproducible training procedure.

In summary, the proposed RNN-based estimator offers an effective, data-driven solution for predicting the optimal step-size in the dual-microphone NLMS framework. By enabling dynamic step-size regulation, the system enhances robustness, accelerates convergence, and improves generalization across a wide range of challenging and non-stationary acoustic scenarios.

4.8.Output Speech Signal Estimation

In this configuration, the adaptive filter $\boldsymbol{w}(n)$is utilized to estimate the acoustic impulse response $\boldsymbol{h}_{21}$ in a dual-microphone convolutive mixing setup. By learning and compensating for the acoustic transfer path between the two sensors, the filter enables the reconstruction of the target speech signal.

After convergence, the optimal value of the adaptive filter can be written as follows—consistent with prior findings reported in [76-79]:

$$\boldsymbol{w}(n) = \boldsymbol{h}_{21} \tag{4.25}$$

With this optimal solution of the adaptive filters, the estimated speech signal can be expressed as follows:

$$\widetilde{sp}(n) = sn(n) * [\boldsymbol{h}_{21} - \boldsymbol{w}(n)] + sp(n) * [\delta(n) - \boldsymbol{h}_{12} * \boldsymbol{w}(n)] \quad (4.26)$$

$$\widetilde{sp}(n) = sp(n) * D_s \quad (4.27)$$

where,$D_s$indicates a minor level of distortion.

4.9.Simulation Study and Performance Results

4.9.1.Acoustic input–Output Signals of the Mixing Model

This section provides a comprehensive simulation study used to assess the performance of the proposed DL-VSS-FNLMS method under different noisy acoustic scenarios. The experiments rely on a realistic two-microphone convolutive mixing framework, presented in chapter 2 (Figure 2.8), which emulates real acoustic conditions by linearly convolving two independent source signals.

- **Source 1 - Clean Speech Signal:**

The first source consists of a clean, phonetically balanced speech signal spoken by a single speaker. The signal is sampled at 8 kHz with 16-bit precision. Figure 4.9 displays the time-domain waveform of this speech signal together with its associated voice-activity-detection (VAD) mask.

The selected utterance is a French sentence with a duration of approximately 4 seconds, extracted from the phonetically balanced speech corpus presented in [80]. The speech data were obtained from the AURORA database [81], which provides standardized speech material for noise-robust processing research.

- **Source 2 — Noise Signal:**

The second source represents a point noise source that emulates real-world acoustic disturbances. Several types of noise were employed, the noise conditions considered in this study include white noise, babble noise, aircraft F-16 noise, industrial factory noise (factory1), HF channel noise, and buccaneer noise. All noise samples are real recordings, acquired at a sampling frequency of 8 kHz and encoded with 16-bit resolution.Figure4.10 provides an illustrative example of the white noise waveform.

The mixture signal reproduces a realistic reverberant acoustic scenario in the presence of environmental noise, is obtained by convolving each source with a distinct room impulse

response, as defined by the acoustical mixing model [82]. These impulse responses are depicted in Figure 4.11characterize the propagation paths between each source and the microphone array in a simulated room setup [83]. This configuration enables the performance evaluation of the developed algorithm under realistic acoustic conditions, including reverberation and multiple noise types.

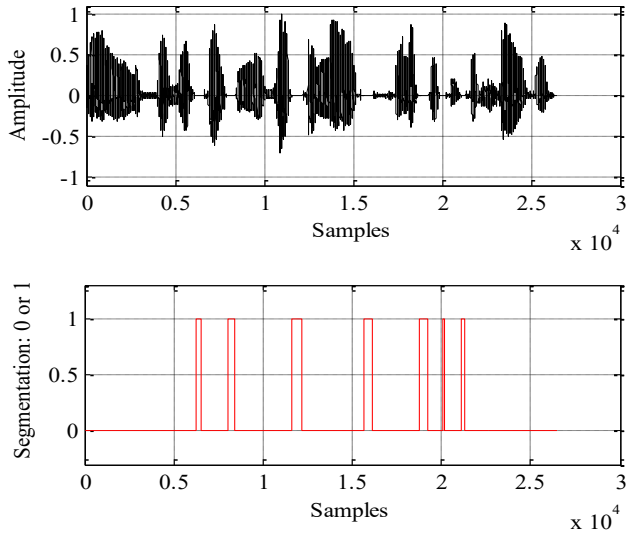

Figure 4.9. Original speech signal and generated segmentation

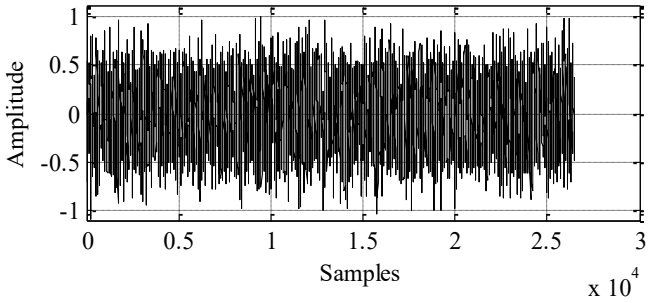

Figure 4.10. Noise signal

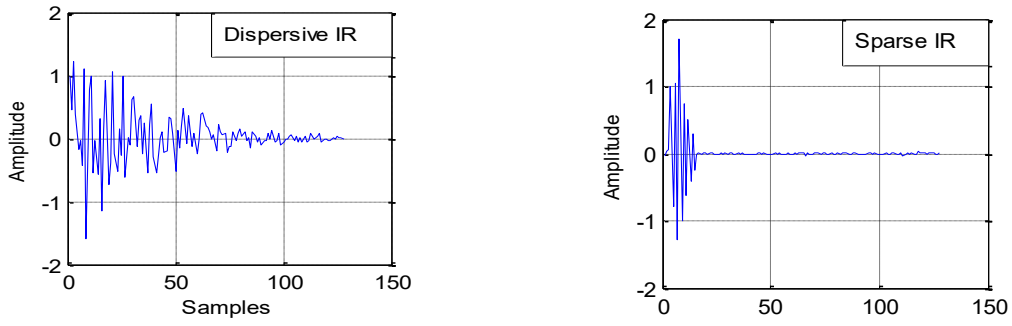

Figure 4.11. Illustrations of practical dispersive and sparse acoustic impulse responses

The noisy mixtures obtained, as shown in Figure 4.12, correspond to an input SNR of –6 dB. This experimental configuration offers a controlled yet realistic environment for assessing the behavior of the adaptive algorithms. It ensures a fair benchmarking process and highlights the superiority of the proposed DL-VSS scheme, particularly in terms of accelerated convergence and improved perceptual speech quality.

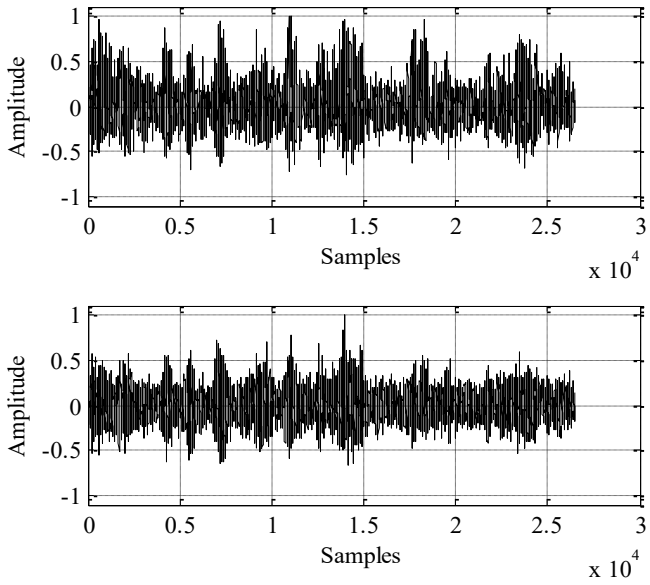

Figure 4.12. Two noisy speech signals with Input-SNR=-6dB

4.3.2. Configuration parameters and performance assessment criteria

Across all simulations conducted in this chapter, the advanced approaches proposed were systematically evaluated using a consistent set of input signals and a controlled range of fixed and adaptive parameters. The selected parameter values were carefully tuned to ensure a fair and comprehensive comparison under various acoustic conditions.

A wide range of noise conditions was considered to simulate realistic and acoustically demanding environments, including white noise, babble noise, F-16 jet recordings, industrial factory noise (factory1), HF channel interference, and buccaneer noise. The acoustic transmission paths were modeled using room impulse responses of equal length for both channels (M = 128), reproducing reverberant propagation between the sound sources and the microphones.

Noisy mixtures were synthesized at several input signal-to-noise ratio (SNR) levels: –6 dB, –3 dB, 0 dB, +3 dB, and +6 dB, with noise applied independently to each microphone signal.

These different SNR settings allowed a comprehensive and systematic evaluation of algorithm resilience under low-, moderate-, and high-noise conditions.

For the proposed DL-VSS-FNLMS algorithm, a diverse collection of internal system parameterswas explored to optimize performance. The maximum allowable step-size parameter, denoted as μ_{max}, was tested with four representative values: 0.5, 0.9, and 1.5. In addition, several critical tuning constants were applied: λ=0.67, ρ=2 and ε=10^{-6}. These parameters play a key role in determining the adaptive filter's behavior, particularly with respect to convergence rate, numerical stability, and noise suppression capability.

The final parameter configuration reflects a balanced trade-off between rapid adaptation and algorithmic stability, avoiding overfitting in highly noisy conditions. All parameter values were determined empirically through multiple simulation rounds conducted on a held-out validation subset using a coarse-to-fine grid search (averaged over several random seeds), and subsequently fixed for final testing.

The performance of the developed algorithmwas assessed within a two-channel convolutive mixing system, tested under both dispersive and sparse impulse response conditions. The evaluation was carried out using the following objective performance assessment criteria:

Time-Domain VSS Evolution:

In this subsection, the temporal evolution of the variable step-size (VSS) parameters estimated by the proposed deep learning model is analyzed. This qualitative examination provides insights into the behavior of the adaptive mechanism over time, particularly its ability to follow signal dynamics and maintain stability during transient periods. The visualization of the VSS trajectory helps assess waveform clarity and the effectiveness of the noise reduction stage in preserving speech structure while suppressing interference.

To provide an objective evaluation of the performance of the deep learning–driven step-size prediction module, three objective metrics are employed:

- **Mean Square Error (MSE):** measures the average squared difference between the predicted and the reference step-size values, reflecting the overall prediction accuracy.

- **Mean Absolute Error (MAE):** computes the mean of absolute deviations, providing a more interpretable measure of average prediction error magnitude.
- **Correlation Coefficient (R^2):** quantifies the degree of linear correlation between predicted and true step-size trajectories, indicating how well the model captures the underlying temporal patterns.

4.9.3. Combined Feature Set and Silence Detection Periods

In this subsection, we present the results derived from integrating the extracted acoustic features with the subsequent silence detection mechanism. To highlight the efficiency of the proposed feature fusion approach as well as the precision of the silence detection, the energy evolution of the combined feature representation is assessed in parallel with the identified silence intervals.

The evaluation was carried out under several challenging acoustic conditions, including signals corrupted by White Gaussian noise, babble noise, F16 aircraft noise, factory noise (Factory1), HF-channel noise, and buccaneer noise. The analysis highlights the robustness of the proposed approach in maintaining consistent feature behavior across diverse noise types.

Figures 4.13 to 4.18 provide graphical illustrations of the feature energy distributions and the corresponding silence detection outcomes for the dispersive system scenario, clearly depicting how the algorithm distinguishes between active speech and non-speech segments under varying noise conditions.

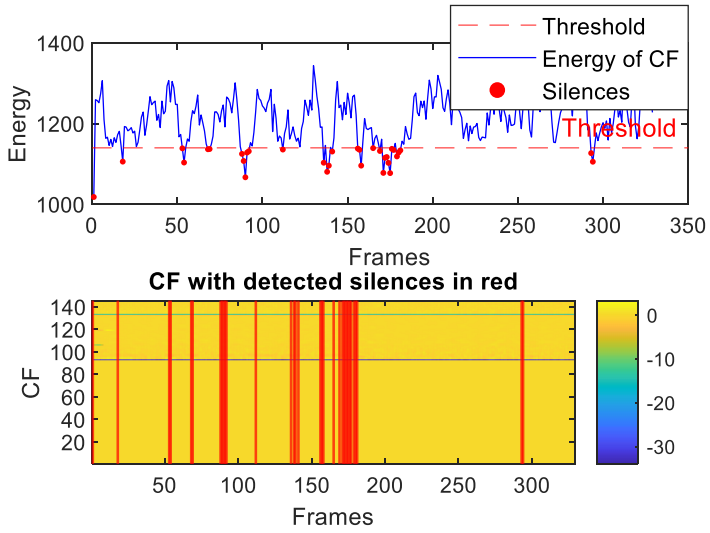

Figure 4.13. Energy of fused features with detected silence periods in dispersive scenario under white noise

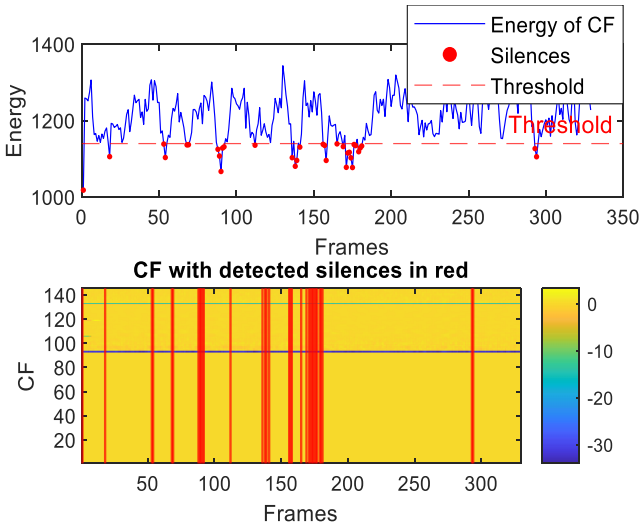

Figure 4.14. Energy of fused features with detected silence periods in dispersive scenario under babble noise

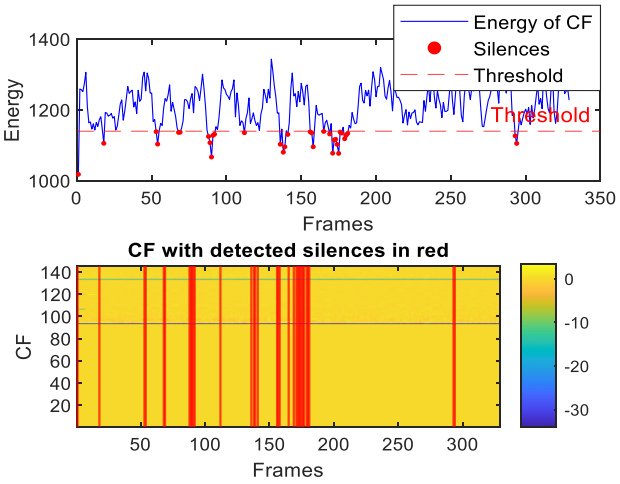

Figure 4.15. Energy of fused features with detected silence periods in dispersive scenario under aircraft noise

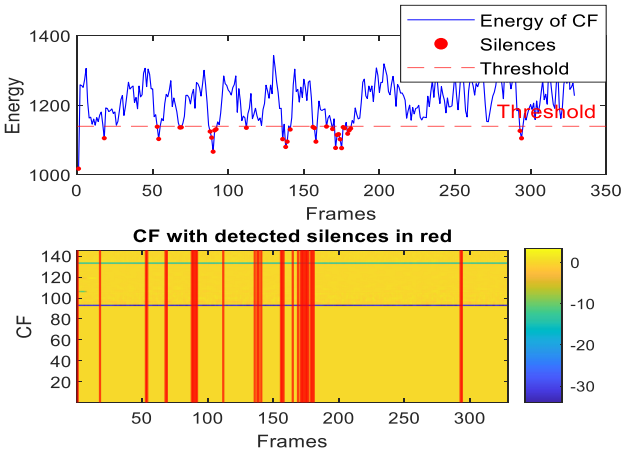

Figure 4.16. Energy of fused features with detected silence periods in dispersive scenario under factory1 noise

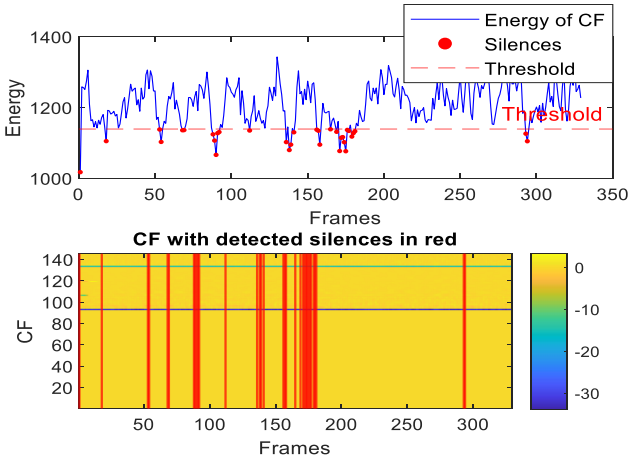

Figure 4.17. Energy of fused features with detected silence periods in dispersive scenario under HfChannel noise

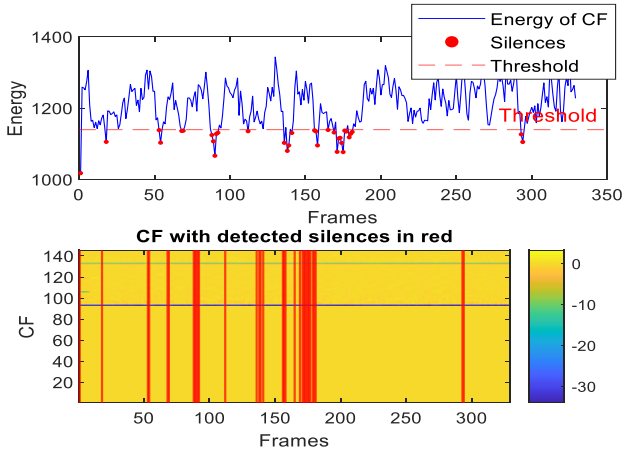

Figure 4.18. Energy of fused features with detected silence periods in dispersive scenario under buccaneer noise

For the dispersive condition (presented inFigures 4.13 to 4.18), a number of key insights emerge regarding the energy evolution of the combined feature set (CF) and the accuracy of the silence detection module.Owing to the spreading effect characteristic of dispersive acoustic environments, the CF energy exhibits noticeable temporal fluctuations. Even minor variations in the input signal (speech + noise) tend to produce broader changes in the corresponding output, highlighting the sensitivity of the energy representation to reverberant propagation effects.

Across Figures 4.13 to 4.18, which correspond to white, babble, F-16 aircraft, factory1, HF-channel, and buccaneer noise environments, the energy contours exhibit dynamic behavior characterized by repeated peaks and troughs. These fluctuations capture the inherent non-stationarity of noisy speech as well as the impact of dispersive room impulse responses.

Importantly, despite these time-varying dynamics, the silence detection module remains highly reliable, with the detected silence segments closely matching low-energy regions across all noise conditions. This confirms the robustness of the proposed energy-based thresholding strategy for precise silence identification, even in strongly dispersive acoustic environments.

In the second phase of the experiments, the evaluation is broadened to include additional acoustic scenarios by integrating sparse acoustic impulse responses, thereby modeling more realistic reverberant environments. The input mixtures are contaminated with the same six noise types used in the dispersive case. Figures 4.19 to 4.24 illustrate the CF energy profiles and the corresponding silence detection outcomes under sparse-reflection conditions, demonstrating the ability of the proposed approach to generalize effectively across diverse acoustic settings.

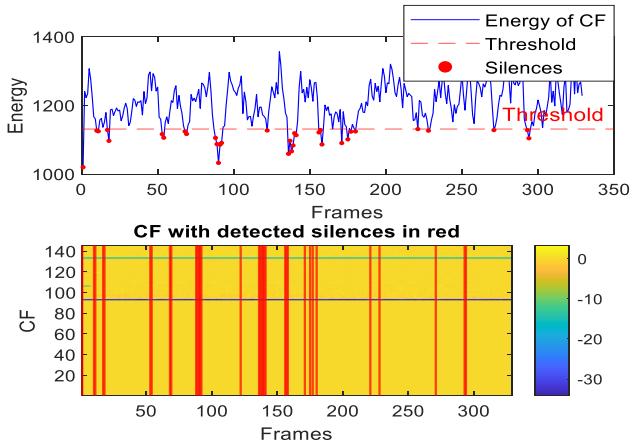

Figure 4.19. Energy of fused features with detected silence periods in sparse scenario with white noise

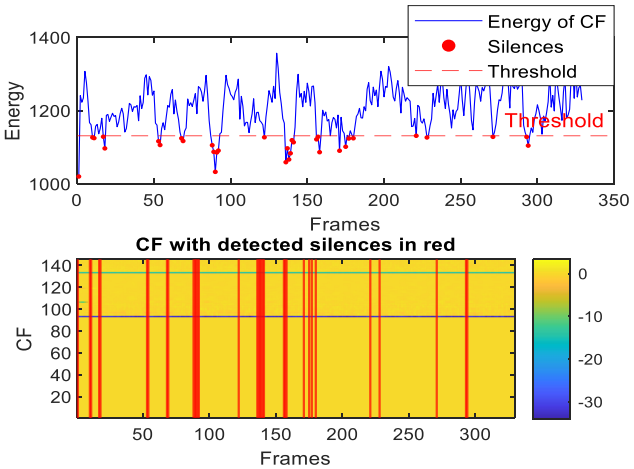

Figure 4.20. Energy of fused features with detected silence periods in sparse scenario with babble noise

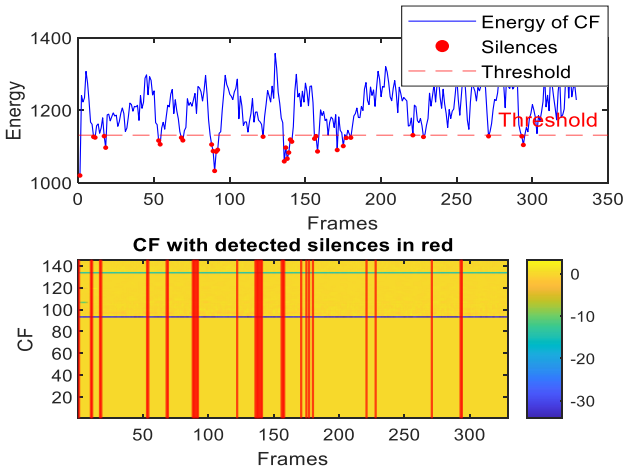

Figure 4.21. Energy of fused features with detected silence periods in sparse scenario with white F16 aircraft noise

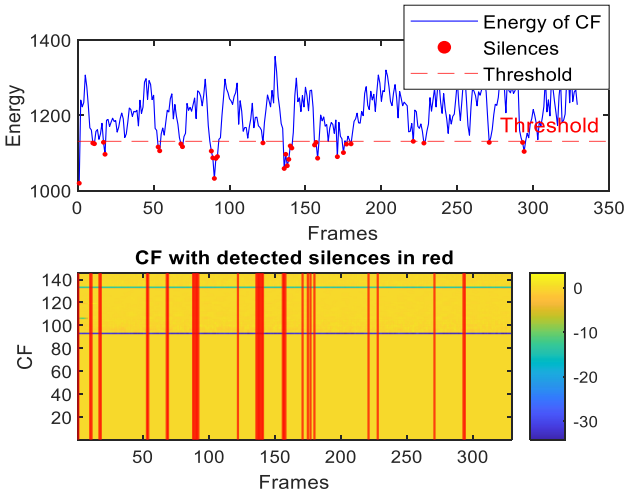

Figure 4.22. Energy of fused features with detected silence periods in sparse scenario with factory1noise

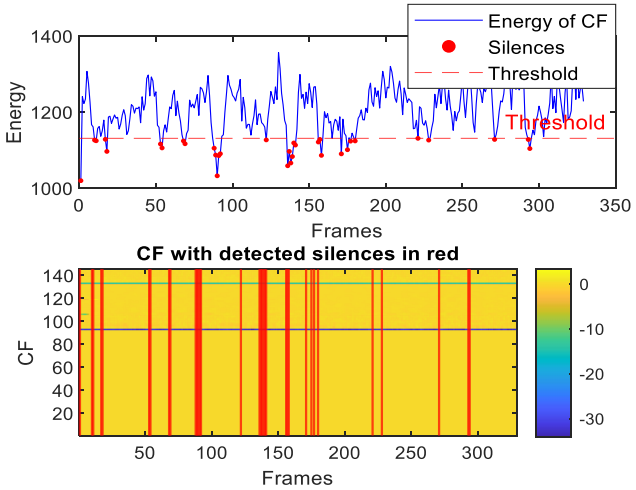

Figure 4.23. Energy of combined features with detected silences periodsin sparse case with Hfchannel noise

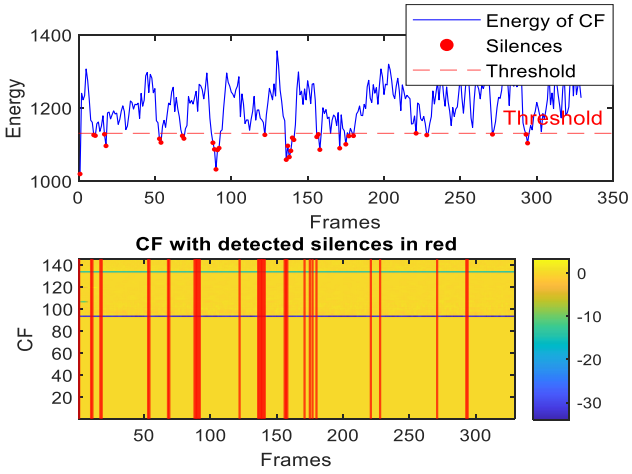

Figure 4.24. Energy of fused features with detected silence periods in sparse scenario with buccaneernoise

In the sparse scenario, the evaluation of the Energy of Combined Features (CF) and the corresponding silence detection performance across various noise environmentsillustratedinFigures 4.19 to 4.24 further confirms the stability and reliability of the developed feature integration and silence identification scheme.

The combined feature (CF) energy displays pronounced temporal dynamics with recurring peaks and troughs, faithfully representing the intrinsic speech patterns present within noisy environments. Such behavior demonstrates the responsiveness of the integrated features to variations in speech presence and acoustic characteristics.

Additionally, the energy-based silence detection scheme reliably pinpoints silent periods by correlating with low-energy regions of the contaminated signal. The consistency of this performance across different noise scenarios highlights the resilience and generalization strength of the proposed detection framework under sparse reverberation conditions.

4.9.4. Deep learning–based estimation of the variable step-size

To examine the effectiveness of the proposed DL-VSS mechanism, its performance was benchmarked against a traditional VSS algorithm under two different acoustic models, namely dispersive and sparse room impulse responses.

For an equitable performance assessment, the two approaches were evaluated under the same configuration, employing three distinct maximum step-size settings (μ_{max} = 0.5, 0.9, and 1.5). Identical speech signals and noise environments were used throughout the experiments, allowing performance differences to be exclusively attributed to the adaptation strategies of each method. This setup enables a clear evaluation of the improvements brought by the DL-

based step-size prediction in terms of stability, convergence rate, and adaptability to varying acoustic dynamics.

For the dispersive case, the obtained results of the step-size variation are presented in Figures 4.25 to 4.27, respectively for three maximal values of step-size, $\mu_{max} = 0.5$, $\mu_{max} = 0.9$ and $\mu_{max} = 1.5$. in other type of impulse response (sparse), the obtained variable step-sizes are presented in Figures 4.28 to 4.30.

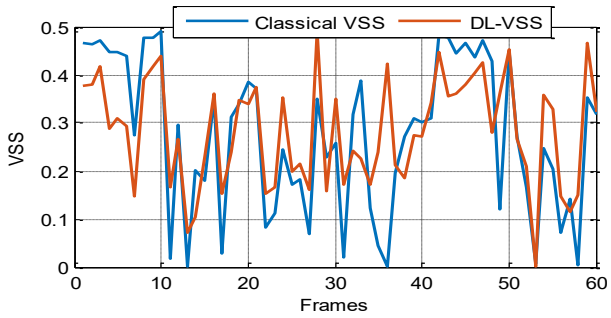

Figure 4.25. Evolution of the DL-based step-size and classical VSS for $\mu_{max} = 0.5$ under dispersive impulse response

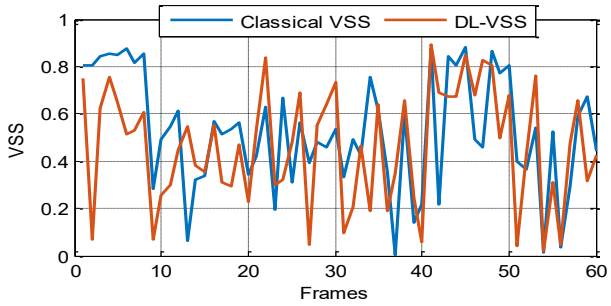

Figure 4.26. Evolution of the DL-based step-size and classical VSS for $\mu_{ax} = 0.9$under dispersive impulse response

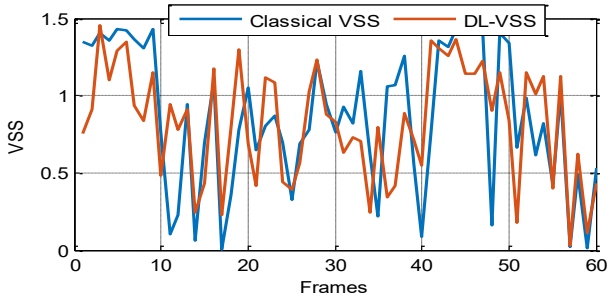

Figure 4.27. Evolution of the DL-based step-size and classical VSS for $\mu_{max} = 1.5$ under dispersive impulse response

As observed in Figures Figures 4.25 to 4.27, the proposed DL-VSS algorithm maintains tracking behavior on par with the traditional VSS scheme under dispersive channel conditions. Furthermore, the performance gap widens in favor of DL-VSS when the maximum step-size value is increased, confirming its superior adaptability in such scenarios.

At a lower maximum step size of $\mu_{max} = 0.5$, the DL-VSS model shows effective tracking behavior, with its adaptation curve appearing smoother and less oscillatory compared to the conventional approach. When the step size is increased to $\mu_{max} = 0.9$,

Under the highly adaptive setting of $\mu_{max} = 1.5$, the superiority of the proposed DL-VSS scheme becomes distinctly apparent. Despite the elevated adaptation gain, the step-size evolution remains well-controlled and does not exhibit the instability frequently observed in traditional VSS algorithms. This result verifies that the learned step-size policy in DL-VSS provides a robust and adaptive regulation strategy, making it especially suitable for dispersive acoustic channels characterized by extended temporal correlations.

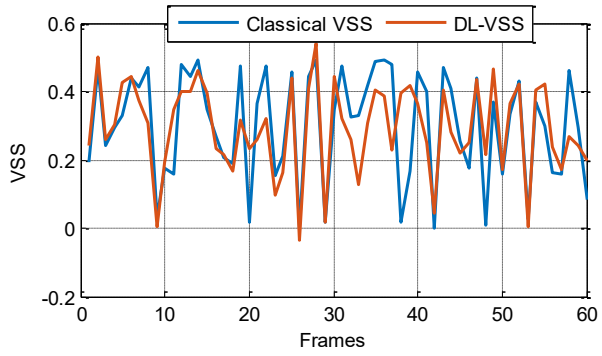

Figure 4.28. Step-size trajectory of DL-VSS and classical VSS for $\mu_{max} = 0.5$ under sparse impulse response

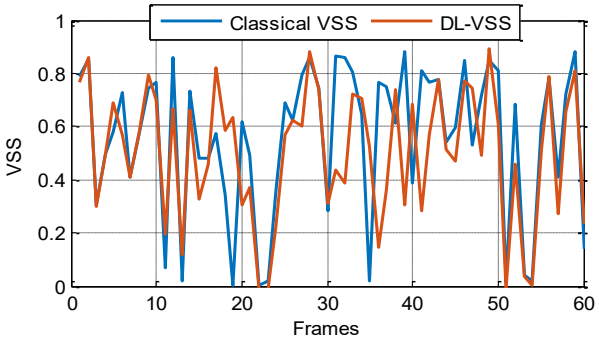

Figure 4.29. Step-size trajectory of DL-VSS and classical VSS for $\mu_{max} = 0.9$ under sparse impulse response

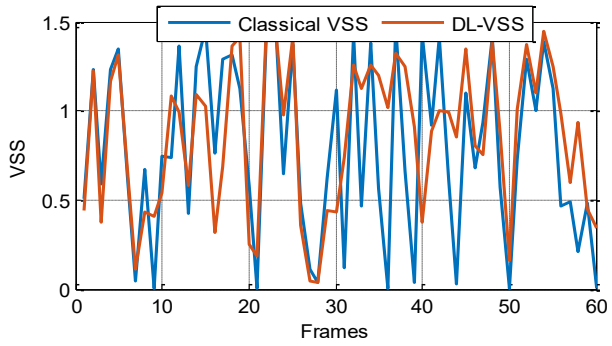

Figure 4.30. Step-size trajectory of DL-VSS and classical VSS for μ_{max} = 1.5 under sparse impulse response

For the sparse impulse-response setting shown in Figures 4.28 to 4.30, the performance comparison between the traditional VSS algorithm and the DL-VSS solution reveals distinct step-size adaptation behaviors for each tested value of μ_{max}.

For μ_{max} = 0.5, the traditional VSS algorithm displays noticeable, abrupt oscillations in its step-size evolution, indicating difficulty in stabilizing around the optimal value. Such irregular behavior reflects challenges in properly tracking the sparse system characteristics and achieving consistent convergence.

Conversely, the DL-VSS variant produces a smoother and more controlled step-size profile, confirming that the learned model effectively exploits the sparse structure of the system to ensure reliable and steady adaptation.

When μ_{max} is increased to 0.9 and 1.5, the DL-VSS continues to operate with significantly reduced instability compared to the classical approach, preserving stable adaptation even under high-gain settings. Across all tested μ_{max} values, the DL-VSS shows consistently regulated and efficient adaptation. These results underline the model's enhanced capability to generalize, adapt, and guarantee stable convergence in sparse acoustic environments, where conventional strategies often suffer from instability and excessive sensitivity to signal fluctuations.

4.9.5. Objective Testing Criteria for the DL Model

A rigorous performance evaluation of the proposed DL-based variable step-size estimator was conducted using a set of objective criteria. These metrics offer quantitative insight into

prediction accuracy and model generalization capability. Specifically, performance was assessed using MSE, MAE, and R^2, which jointly characterize the degree of similarity and error between the DL-predicted step-size values and the classical VSS reference.

Table 4.1 presents the quantitative results for these indicators over various input SNR conditions under both dispersive and sparse acoustic environments. The consistently favorable values achieved across all scenarios confirm the accuracy, stability, and robustness of the proposed predictor, demonstrating strong generalization to different acoustic propagation characteristics.

Table 4.1.DL-model validation results using MAE, MSE, and R^2 across multiple SNR conditions in dispersive and sparse cases (three input SNR values)

Types of IR	Input SNR in dB	Testing Criteria		
		MAE	MSE	R^2
Dispersive case	-6	0.1448	0.0521	0.1204
	-3	0.1784	0.0516	0.1403
	0	0.1913	0.0605	0.1800
	3	0.0504	0.1787	0.1892
	6	0.1532	0.0402	0.3949
Sparse case	-6	0.1291	0.0384	0.2995
	-3	0.1571	0.0523	0.3613
	0	0.1450	0.0460	0.4760
	3	0.1154	0.0251	0.6514
	6	0.1448	0.0434	0.3955

In the dispersive case, the DL-VSS algorithm achieves its highest predictive accuracy at an input SNR of 3 dB, where it records the lowest MAE (0.0504) and a notably low MSE (0.1787). This indicates that the predicted step-size parameters are most closely aligned with the reference values under moderately noisy dispersive conditions. However, in the dispersive configuration, the correlation coefficient R^2 remains relatively low for all evaluated SNR conditions, with the highest value reaching only 0.3949 at 6 dB. Particularly low R^2 scores are observed at challenging noise levels such as –6 dB (0.1204) and –3 dB (0.1403). These results indicate that although the model attains acceptable error values at certain SNRs, its ability to

account for the variability of the true step-size trajectory in dispersive environments is limited. This behavior likely stems from the difficulty of capturing long-range temporal dependencies and highly correlated acoustic patterns in severe noise scenarios, which characterizes dispersive channels.

In contrast, the sparse acoustic case exhibits markedly stronger and more stable performance across the tested SNR range. The model achieves its best accuracy at 3 dB SNR, obtaining the lowest MAE (0.1154) and MSE (0.0251). These values consistently outperform those recorded in the dispersive setup at nearly every SNR, suggesting that the model efficiently exploits the inherent sparsity of the system to estimate the step-size more accurately.

Moreover, the R^2 scores in the sparse condition are substantially higher, peaking at 0.6514 at 3 dB SNR. This elevated correlation demonstrates a much stronger correspondence between the predicted and ground-truth step-size values, confirming the model's high predictive capability under moderate noise levels. Even at –6 dB, the model maintains a comparatively higher R^2 value (0.2995), further illustrating its enhanced resilience and ability to extract meaningful structure in sparsely reverberant environments.

To reinforce these findings, a comparative evaluation between the proposed DL-VSS-FNLMS approach and the classical FNLMS algorithm with fixed step-sizes was conducted. This analysis, carried out under both dispersive and sparse acoustic conditions, aims to verify the generalization and robustness of the proposed learning-based solution when dealing with varying room impulse response characteristics.

4.10. Conclusion

In this chapter, we introduced a significant advancement in acoustic noise reduction with the Deep Learning–based Variable Step-Size (DL-VSS) NLMS algorithm, integrating it into a two-sensor feed-forward architecture. The fundamental challenge addressed was the inherent compromise between fast convergence and low steady-state error in traditional adaptive filters, especially within complex sparse and dispersive acoustic environments. Our innovation leveraged a RNN, specifically using stacked LSTM layers, trained on a diverse set of acoustic features (MFCCs, GTCCs, ERB, Bark, and Mel). This neural network was successfully employed as an intelligent step-size predictor, offering a data-driven approach to dynamically modulate the learning rate in real time. The integration of a reliable silence

detection mechanism further enhanced the model's context-awareness and predictive robustness across various noise conditions, ensuring stable and precise step-size control.

Objective performance evaluation confirmed the superiority and robustness of the proposed DL-VSS-FNLMS. Comparative analysis against the classical Variable-Step-Size FNLMS algorithm demonstrated marked improvements across critical performance metrics. Specifically, the deep learning-driven adaptation strategy yielded significantly faster convergence, leading to rapid noise suppression, alongside a lower steady-state values, which directly contributes to enhanced fidelity. The adaptive control provided by the DL model effectively mitigated the risk of the large step-size instabilities often encountered by conventional VSS methods, solidifying its effectiveness as a reliable and high-performance solution for real-world acoustic challenges.

General Conclusion

The research presented in this thesis has focused on the development of advanced adaptive filtering techniques for acoustic noise reduction and speech enhancement in complex and realistic environments.The main motivation of this work stemmed from the limitations of conventional adaptive algorithms such as LMS and NLMS, which, despite their simplicity and wide adoption, exhibit a critical trade-off between convergence speed and steady-state accuracy, particularly under non-stationary and dispersive acoustic conditions.The objective was therefore to design new two-sensor adaptive algorithms that combine the flexibility of traditional adaptive filters with the learning and generalization capabilities of neural networks and deep learning architectures.

The first part of this thesis provided a comprehensive overview of adaptive filtering theory, including the principles of the Wiener filter and stochastic gradient algorithms. These classical methods were analyzed in the context of speech and noise modeling, establishing a strong theoretical foundation for understanding adaptive signal processing systems.

The second part explored BSS as a powerful framework for noise cancellation and speech enhancement. Two main structures were implemented and analyzed: the Forward and Backward configurations, both applied to two-sensor convolutive mixtures of speech and noise. Several adaptive algorithms were evaluated in this context, such as the Two- sensor LMS (2LMS), Two-sensor NLMS (2NLMS), and SAD. The comparative analysis demonstrated that the Backward structure provided the best speech reconstruction with minimal distortion, while the Forward structure achieved faster convergence and stronger noise reduction.These findings confirmed the potential of BSS-based models as efficient solutions for acoustic noise reduction in real environments.

Building on these results, the third part of the thesis introduced a novel Neural Network-based Variable Step-Size Feed-forward NLMS (NN-V-FNLMS) algorithm.This method integrated a simple neural network capable of learning the relationship between signal statistics and optimal adaptation parameters, enabling a dynamic adjustment of the step-size at each iteration.

By incorporating a VAD, the adaptation process was selectively activated during noise-only periods, thereby reducing computational complexity and improving efficiency. Simulation

results revealed that the proposed NN-V-FNLMS algorithm significantly improved convergence rate, tracking capability, and speech quality compared to conventional LMS/NLMS and classical VSS algorithms. This contribution demonstrated that even a shallow neural network can provide effective non-linear modeling of step-size dynamics, bridging the gap between classical signal processing and modern machine learning.

In the final part, a new DL-VSS-FNLMS algorithm was proposed.Thisalgorithm incorporated a RNN architecture with LSTM layers, trained to estimate the optimal step-size dynamically from a set of acoustic features including MFCCs, GTCCs, and spectral representations on the ERB, Bark, and Mel scales.

The deep learning model was trained on realistic noisy speech datasets generated using a two-channel convolutive mixing model under both dispersive and sparse impulse response conditions.Experimental evaluations demonstrated that the DL-VSS-FNLMS algorithm not only improved the accuracy of step-size prediction but also provided faster convergence, better system tracking, and higher segmental SNR values. These results validated the effectiveness of deep learning in adaptively controlling the filtering process, making it possible to achieve real-time, data-driven adaptation in highly dynamic acoustic environments.

Perspectives and Future Work

While the results obtained in this work are very encouraging, several research directions can be pursued to further extend and enhance the proposed methodologies:

- Implementing the proposed DL-VSS-FNLMS algorithm in sub-band or frequency-domain form to reduce computational load and improve convergence in wideband applications.
- Generalizing the two-sensor framework to multi-microphone systems and integrating beamforming techniques for spatial filtering and directional noise suppression.
- Exploring transformer-based or graph neural network (GNN) architectures to capture long-range temporal dependencies and spatial relationships between microphones.
- Developing optimized real-time implementations on embedded systems or digital signal processors (DSPs) to validate practical feasibility in low-power, real-world applications.

REFERENCES

[1] Wang, D., Chen, J., « Deep Learning for Speech Enhancement », Ouvrage, Springer, New York, 2020.

[2] Kim, S., Park, J., Lee, K., « Deep Learning-Based Speech Enhancement », Ouvrage, IEEE Press, 2021.

[3] Defossez, A., Synnaeve, G., Adi, Y., « Hybrid Neural Speech Enhancement », Rapport technique, Meta AI Research, 2023.

[4] Haykin, S., « Adaptive Filter Theory », Ouvrage, Prentice Hall, New Jersey, 2002.

[5] Oppenheim, A. V., Schafer, R. W., « Digital Signal Processing », Ouvrage, Prentice Hall, 2010.

[6] Li, Z., Huang, X., Zhang, Y., « Variable Step Size LMS with Momentum », Article, IEEE Transactions on Signal Processing, 2023.

[7] Zhu, L., Wang, M., Liu, J., « p-Power Multi-Moment LMS Algorithm », Article, Signal Processing, 2023.

[8] Soon, I.-Y., Tong, S., et al., « Variable Step-Size LMS Algorithms », Article, IEEE Signal Processing Letters, 1998.

[9] Abd El-Fattah, M., Abdelhamid, A., Mansour, A., « Advanced Variable Step-Size LMS Algorithms », IEEE Signal Processing Magazine, 2023.

[10] Chen, B., Zhao, H., Principe, J. C., « Modern Convergence Analyses for Adaptive Algorithms », IEEE Transactions on Signal Processing, 2024.

[11] Pascual, S., Serrà, J., Bonafonte, A., 'Towards Generalized Speech Enhancement with Generative Adversarial Networks,' in Proc. INTERSPEECH, 2019, pp. 1791–1795, doi: 10.21437/Interspeech.2019-2688.

[12] Zhang, L., Xu, Y., Li, P., « RNN-Based Speech Enhancement Methods », Article, IEEE/ACM Transactions on Audio, Speech and Language Processing, 2021.

[13] Tan, K., Wang, D., « Complex Spectral Mapping for Speech Enhancement », Article, IEEE/ACM TASLP, 2021.

[14] Mawalim, M., Rahman, A., Yosida, H., « Deep Learning-Based Speech Enhancement in Real-World Conditions », Article, Applied Acoustics, 2024.

[15] Xie, Z., Tan, K., « Complex Spectrograms for Speech Enhancement: A Survey », Revue, IEEE Signal Processing Magazine, 2025.

[16] Mensah, P., Adusei, S., Boateng, E., « Deep Learning Speech Enhancement for Robust Classification », Article, Neural Networks, 2025.

[17] Van Gerven, S., Van Compernolle, D., 'Feed forward and Feed back in a symmetric adaptive noise canceller: stability analysis in a simplified case', Proc. IEEE EUSIPCO, Belgium, Brussels, V.1, (Aug. 1992), pp. 1081–1084.

[18] Belouchrani, A. (1995). Blind source separation: Algorithms, performance analysis and application to experimentalsignals (Doctoral dissertation). École Nationale Supérieure des Télécommunications, Paris, France.

[19] Mansour, A. (1997). Contribution to blind source separation (Doctoral dissertation). Institut National Polytechnique de Grenoble, Grenoble, France.

[20] Amehraye, A. (2009). Perceptual speech denoising (Doctoral dissertation). École Nationale Supérieure des Télécommunications de Bretagne, France.

[21] Jutten, C. (2009). Théorie du signal. Université Joseph Fourier – Grenoble INP, France.

[22] Wiener, N., 'Extrapolation, interpolation and smoothing of stationary time series', John Wiley & Sons, New York, 1949.

[23] Bellenger, M., 'Traitement numérique du signal', 2ème édition, MASSON, 1987.

[24] Vaseghi, S. V., 'Advanced digital signal processing and noise reduction', Second Edition, John Wiley & Sons, 2000.

[25] Manolakis, D. G., Ingle, V. K., Kogon, S. M., 'Statistical and adaptive signal processing', Artech House, 2005.

[26] Bellanger, M., 'Traitement numérique du signal théorie et pratique', 8ème édition, Dunod, 2006.

[27] Diniz, P. S. R., 'Adaptive filtering algorithms and practical implementation', Second Edition, Springer, 2008.

[28] Widrow, B., Hoff, M. E., 'Adaptive switching circuits', WESCOM Conv. Rec., V.4, (1960), pp. 96–140.

[29] Widrow, B., McCool, J. M., Larimore, M. G., Johnson, C. R., 'Stationary and nonstationary learning characteristics of the LMS adaptive filters', Proceedings of the IEEE, V.64, (Aug. 1976), pp. 1151–1162.

[30] Riegler, R., Compton, R., 'An adaptive array for interference rejection', Proc. IEEE, V.61, (Jun. 1973), pp. 748–758.

[31] Haykin, S., Adaptive Filter Theory, 4th edition, Prentice Hall, Upper Saddle River, New Jersey, 2002.

[32] Alaeddine, H., 'Application de la transformée en nombres entiers à la conception d'algorithmes de faible complexité pour l'annulation d'échos acoustiques', Thèse de Doctorat, Université de Bretagne occidentale, (Jul. 2007).

[33] Ozeki, K., Umeda, T., 'An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties', Elec. Comm. Japan, V.J67-A, (Feb. 1984), pp. 126–132.

[34] Berouti, M., Schwartz, R., Makhoui, J., 'Enhancement of speech corrupted by acoustic noise', Proc. ICASSP, Washington, United States, (1979), pp. 208–211.

[35] Lockwood, P., Boudy, J., 'Experiments with a nonlinear spectral subtractor (NSS), hidden Markov models and the projection, for robust speech recognition in cars', Speech Communication, V.11, n° 2-3, (1992), pp. 215–228.

[36] Mokbel, C., Barbier, L., Chollet, G., 'Adapting a HMM speech recognizer to noisy environments', Workshop on Speech Processing in Adverse Conditions, Cannes, (1992), pp. 211–214.

[37] Compernolle, D. V., 'Noise adaptation in a hidden Markov model speech recognition system', Computer Speech and Language, V.3, (1989), pp. 151–167.

[38] Ephraim, Y., Malah, D., 'Speech enhancement using a minimum mean square error short-time spectral amplitude estimator', IEEE Trans. ASSP, V.32, n° 6, (1984), pp. 1109–1121.

[39] Howells, P., 'Intermediate frequency side-lobe canceller', US patent 3202 990, (Aug. 1965).

[40] Buchner, H., Benesty, J., Kellermann, W., 'Generalized multichannel frequency-domain adaptive filtering: efficient realization and application to hands-free speech communication', Signal Processing, V.85, (2005), pp. 549–570.

[41] Souden, M., Benesty, J., Affes, S., 'On the global output SNR of the parameterized frequency-domain multichannel noise reduction Wiener filter', IEEE Signal Processing Letters, V.17, n° 5, (May 2010), pp. 425–428.

[42] Spriet, A., 'Adaptive Filtering Techniques for Noise Reduction and Acoustic Feedback Cancellation in Hearing Aids', PhD Thesis, Katholieke University Leuven, Belgium, (Sep. 2004).

[43] Jutten, C., Herault, J., 'Blind separation of sources, part I: an adaptive algorithm based on neuromimetic architecture', Signal Processing, V.24, n° 1, (Jul. 1991), pp. 1–10.

[44] Boumaraf, H., 'Séparation aveugle de mélanges convolutifs de sources', Thèse de Doctorat, Université Joseph Fourier, INPG, France, 2005.

[45] Van Gerven, S., 'Adaptive noise cancellation and signal separationwith applications to speech enhancement', Ph.D. dissertation, Université Catholique de Leuven, (Mar. 1996).

[46] Djendi, M., Gilloire, A., Scalart, P., 'Noise cancellation using two closely spaced microphones: experimental study with a specific model and two adaptive algorithms', IEEE Int. Conf. ICASSP, Toulouse, France, V.3, (May 2006), pp. 744–748.

[47] Djendi, M., 'Advanced techniques for two-microphone noise reduction in mobile communications', Thèse de Doctorat, Université de Rennes 1, France, (Jan. 2010).

[48] Bendoumia, R., 'Annulation du bruit par les méthodes de séparation de sources aveugles. Application aux systèmes de télécommunications numériques', Thèse de Doctorat, Université de Blida 1, Algérie, 2014.

[49] Parra, L., Spence, C., 'Convolutive blind source separation of non-stationary sources', IEEE Transactions on Speech and Audio Processing, V.8, n° 3, (May 2000), pp. 320–327.

[50] Knezevic, D., 'Blind source separation for signal processing applications', Ph.D. dissertation, University of Western Australia, 2004.

[51] Mukai, R., Araki, S., Makino, S., 'Separation and dereverberation performance of frequency domain blind source separation', in Proc. Independent Component Analysis (ICA'01), (Dec. 2001), pp. 230–235.

[52] Feder, M., Oppenheim, A. V., Weinstein, E., 'Maximum likelihood noise cancellation using the EM algorithm', IEEE Trans. on Acoustics, Speech and Signal Processing, V.ASSP-37, n° 2, (Feb. 1989).

[53] Harrison, W. A., Lim, J. S., Singer, E., 'A new application of adaptive noise cancellation', IEEE Trans. on Acoustics, Speech and Signal Processing, V.ASSP-34, n° 1, (Feb. 1986).

[54] Alkindi, M. J., Dunlop, J., 'Improved adaptive noise cancellation in the presence of signal leakage on the noise reference channel', Signal Processing, V.17, (1989), pp. 241–250.

[55] Nguyen Thi, H. L., Jutten, Ch., Caelen, J., 'Séparation aveugle de parole et de bruit dans un mélange convolutif', Treizième colloque Gretsi, Juan-les-Pins, (Sep. 1991).

[56] Vallauri, A., 'L'étude et le développement de méthodes de reconnaissance de la parole et de réduction du bruit, et application', Thèse de Doctorat, Université de Nice, France, (Dec. 1992).

[57] Van Gerven, S., Van Compernolle, D., 'On the use of decorrelation in scalar signal separation', In: Proc. ICASSP, Adelaide, South Australia, V.3, (1994), pp. 57–60.

[58] Van Gerven, S., Van Compernolle, D., 'Signal separation by symmetric adaptive decorrelation: Stability, convergence and uniqueness', IEEE Transactions on Signal Processing, V.43, n° 7, (Jul. 1995), pp. 1602–1612.

[59] Bendoumia, R., Deba, A., 'Rehaussement du signal de parole par l'algorithme de décorrélation symétrique', Thèse de Master, Dépt. électronique, Université de Blida, (Jul. 2011).

[60] Ikeda, S., Sugiyama, A., 'An adaptive noise canceller with low signal distortion in the presence of crosstalk', IEICE Trans. Fundamentals, V.E.82-A, n° 8, (Aug. 1999), pp. 1517–1525.

[61] Bendoumia, R., 'New two-microphone simplified sub-band forward algorithm based on separated variable step-sizes for acoustic noise reduction', Applied Acoustics, vol. 222, 110069, 2024.

[62] Cheng, G., Liao, L., Chen, K., Hu, Y., Zhu, C., Lu, J., 'Semi-blind source separation using convolutive transfer function for nonlinear acoustic echo cancellation', J. Acoust. Soc. Am., vol. 153, pp. 88–95, 2023.

[63] Van Gerven, S., 'Adaptive noise cancellation and signal separation with applications to speech enhancement', Ph.D. dissertation, UniversitéCatholique de Leuven, Mars 1996.

[64] Bendoumia, R., 'New sub-band proportionate forward adaptive algorithm for noise reduction in acoustical dispersive-and-sparse environments', Applied Acoustics, vol. 175, 107822, 2021.

[65] Van Gerven, S., Van Compernolle, D., 'Signal Separation by Symmetric Adaptive Decorrelation: Stability, Convergence, and Uniqueness', IEEE Transactions on Signal Processing, vol. 43, pp. 1602–1612, 1995.

[66] Charkani, N. H., 'Auto-adaptive separation of convolutive mixtures, Applications to hand-free telephony in cars', Ph.D. dissertation (in French), Institut National Polytechnique de Grenoble, France, 1996.

[67] Bendoumia, R., 'Advanced Feedforward-and-Feedback Decorrelation Algorithms for Speech Quality Enhancement', Inter. Conf. on Electrical Engineering and Control Applications, Springer, 2019, pp. 1157–1170.

[68] Apolinário, J. A., Rautmann, R. (2009). QRD-RLS adaptive filtering.In J. A. Apolinário (Ed.).Springer New York, USA, pp. 1–350.

[69] Combescure, P., '20 listes de dix phrases phonétiquement équilibrées', 1981.

[70] Hirsch, H. G., 'The Aurora experimental framework for the performance evaluations of speech recognition systems under noisy conditions', ISCA ITRW ASR, Automatic Speech Recognition: Challenges for the Next Millennium, 2000.

[71] Hassani, I., Bendoumia, R., Guessoum, A., Abed, A. (2024). New Variable selected coefficients adaptive sparse algorithm for acoustic system identification. Traitement du Signal, 41(3): 1089–1099.

[72] Duttweiler, D. L. (2002). Proportionate normalized least mean-squares adaptation in echo cancelers. IEEE Transactions on Speech and Audio Processing, 8(5): 508–518.

[73] Benesty, J., Gay, S. L. (2002). An improved PNLMS algorithm. In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Orlando, FL, USA, 2: II-1881.

[74] Zue, V., Seneff, S., Glass, J. (1990). Speech database development at MIT: TIMIT and beyond. Speech Communication, 9(4): 351–356.

[75] Varga, A., Steeneken, H. J. (1993). Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. Speech Communication, 12(3): 247–251.

[76] Kolbæk, M. (2018). Single-microphone speech enhancement and separation using deep learning.AALBORG Universitet.

[77] Bendoumia, R., Djendi, M. (2015). Two-channel variable-step-size forward-and-backward adaptive algorithms for acoustic noise reduction and speech enhancement. Signal Processing, 108: 226–244.

[78] Liao, C. F., Tsao, Y., Lee, H. Y., Wang, H. M. (2019). Noise adaptive speech enhancement using domain adversarial training. Proceedings of the Annual Conference of The International Speech Communication Association, Interspeech, 2019: 3148–3152.

[79] Cheng, G., Liao, L., Chen, K., Hu, Y., Zhu, C., Lu, J. (2023). Semi-blind source separation using convolutive transfer function for nonlinear acoustic echo cancellation. The Journal of the Acoustical Society of America, 153(1): 88–95.

[80] Hassani, I., Arezki, M., Benallal, A. (2020). A novel set membership fast NLMS algorithm for acoustic echo cancellation. Applied Acoustics, 163: 107210.

[81] Pearce, D., Hirsch, H. G. (2000). The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions.In 6th International Conference on Spoken Language Processing, Beijing, China, pp. 29–32.

[82] Al-Kindi, M. J., Dunlop, J. (1989). Improved adaptive noise cancellation in the presence of signal leakage on the noise reference channel.Signal Processing, 17(3): 241–250.