

P.N 001 / 01

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**ECOLE NATIONALE POLYTECHNIQUE**  
**DEPARTEMENT D'ELECTRONIQUE**



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

**PROJET DE FIN D'ETUDES**  
**POUR L'OBTENTION DU DIPLOME D'INGENIEUR D'ETAT EN ELECTRONIQUE**



*Mise en œuvre d'un annuleur  
d'écho acoustique*

**Proposé et dirigé par :**

**M<sup>r</sup> A. BELOUHRANI**  
**M<sup>r</sup> A. MOKRAOUI**

**Etudié par :**

**M<sup>r</sup> F. ABDA**  
**M<sup>r</sup> S. MOUSSAOUI**

**Promotion : Juin 2001**

**E.N.P 10, Avenue Hassen BADI, B.P.182 EL-HARRACH, ALGER, ALGERIE**

A mes chers parents, à ma chère tante Amel, à mon cher frère Amir et à toute ma famille sans exception aucune. A tous mes amis.

Farès ABDA

A mes chers parents, à ma soeur Ghalia, à mes frères Djemel, Farid, Mustapha, Hamid et Belkacem, à mes grand-mères et à tous le reste de ma famille.  
A tous mes copains et copines.

Saïd MOUSSAOUI

## *Remerciements*

Premièrement nous tenons à exprimer notre sincère gratitude à nos superviseurs, M<sup>r</sup> A. Belouchrani pour ses conseils fructueux, et M<sup>r</sup> A. Mokraoui pour l'assistance constante qu'il a bien voulu nous apporter sans lésiner aucune attention.

Nous tenons également à remercier M<sup>me</sup> Hamami et M<sup>r</sup> Berkani ainsi que M<sup>elle</sup> Merazka pour l'aide précieuse qu'ils ont bien voulu nous apporter ainsi que M<sup>r</sup> Bousseksou pour avoir bien voulu présider le jury.

Sans oublier au passage Youcef d'Alger plage, Saliha de l'INELEC, Okba de Sour-El-Ghozlane et tous celles et ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.

à tout le monde, merci encore et encore !

## Abstract

In this work, an algorithm is applied for adaptive filtering in the frequency-domain with application to acoustic echo cancellation. This algorithm, called generalized multi-delay filter (GMDF), is derived from the frequency-domain implementation of the time-domain block least mean square algorithm. Some control strategies are implemented ; in particular a double-talk detector based on a spectral dissimilarity measure is introduced ; also a method for single channel noise reduction is presented, based on the spectral subtraction technique with its extension to residual acoustic echo suppression.

## ملخص

في هذا العمل، نستعمل خوارزمية للترشيح التكيف في المجال التوتري مع تطبيقها لإلغاء الصدى الصوتي. هذه الخوارزمية و المسماة المرشح المتعدد المدى المعمم (GMDF)، هذا الأخير مستنبط من تطبيق خوارزمية المعدل التربيعي الأصغر في المجال التوتري.

نلجأ إلى بعض التحسينات و بصفة خاصة نستعمل مبرزا للإتصال المزدوج يعتمد على قياس عدم التشابه التوتري. بالإضافة إلى تنقيص شدة الضجيج بواسطة تقنية الطرح التوتري مع تعميمها لغاية إلغاء الصدى الصوتي المتبقي.

## Résumé

Dans ce travail, un algorithme de filtrage adaptatif dans le domaine fréquentiel est appliqué à l'annulation de l'écho acoustique. Cet algorithme, appelé filtre généralisé à délais multiples (GMDF), est dérivé de l'implémentation de l'algorithme du gradient stochastique en blocs dans le domaine fréquentiel. Quelques stratégies d'amélioration sont implémentées, en particulier un détecteur de double activité vocale fondé sur la mesure de dissimilarité spectrale est introduit, d'autre part, une méthode pour la réduction de bruit à canal unique est présentée, basée sur la technique de soustraction spectrale avec son extension à la suppression de l'écho résiduel.

## Mots clefs

Filtrage adaptatif, Convolution rapide dans le domaine fréquentiel, Annulation d'écho, Communication mains-libres, Double activité vocale, Soustraction spectrale, Réduction de bruit, Statistiques minimales.

# Sommaire

## Introduction générale

<b>1. Introduction à l'annulation d'écho acoustique</b>	<b>1</b>
1.1 Introduction	1
1.2 L'écho acoustique	2
1.3 Mécanisme de couplage rétroactif	3
1.4 Annulation de l'écho acoustique par filtrage adaptatif (AE)	4
1.5 Modélisations linéaires	5
1.5.1 Modèle à Réponse Impulsionnelle Finie (RIF)	7
1.5.2 Modèle Auto Regressif avec entrée eXterne (ARX)	8
1.5.3 Modèle OE (Output Error)	9
1.6 Structures algorithmiques	11
1.6.1 Structures dérivées de l'algorithme du gradient	11
1.6.2 Algorithmes des moindres carrés recursifs	11
1.6.3 Algorithme de Newton rapide	12
1.6.4 Algorithmes de filtrage adaptatif en sous-bandes	12
1.7 Annulation de l'écho stéréophonique	13
1.8 Conclusion	13
<b>2. Algorithmes de filtrage adaptatif</b>	<b>15</b>
2.1 Introduction au filtrage adaptatif	16
2.1.1 Structure d'un filtre adaptatif	16
2.2 Algorithmes de filtrage adaptatif dans le domaine temporel	20
2.2.1 Algorithme LMS	20
2.2.2 Algorithme NLMS	22
2.2.3 Algorithme BLMS	24

2.3	Algorithmes de filtrage adaptatif dans la domaine fréquentiel	
2.3.1	Algorithme FBLMS	26
2.3.2	Algorithme MDF	30
2.3.3	Algorithme GMDF $\alpha$	34
2.4	Implémentation de l'algorithme GMDF $\alpha$	38
2.5	Etude de la complexité arithmétique de l'algorithme GMDF $\alpha$	39
2.6	Conclusion	40
<b>3.</b>	<b>Simulations et résultats</b>	<b>41</b>
3.1	Modèle de simulations et notations	41
3.2	Grandeurs décrivant les performances	42
3.2.1	L'erreur quadratique moyenne	42
3.2.2	L'ERLE (Echo Return Loss Enhancement)	42
3.3	Analyse dans un contexte stationnaire	43
3.4	Analyse dans un contexte non stationnaire	48
3.4.1	Analyse avec des signaux de parole	48
3.4.2	Effet d'une variation du système à identifier	55
3.5	Conclusion	56
<b>4.</b>	<b>Application avec des signaux expérimentaux</b>	<b>57</b>
4.1	Notations	57
4.2	Mécanisme expérimental d'acquisition des signaux	58
4.3	Résultats expérimentaux	58
4.3.1	Application dans le cas "simple duplex"	59
4.3.2	Application dans le cas "full duplex"	64
4.3.3	Application dans le cas "full duplex" avec présence du bruit	65
4.4	Conclusion	69
<b>5.</b>	<b>Amélioration des performances</b>	<b>70</b>
5.1	Introduction	70
5.2	Analyse de la suppression soustractive du bruit	71
5.2.1	Modélisation du bruit additif	71
5.2.2	Estimateur de soustraction spectrale	72
5.2.3	Rectification demi-onde	72
5.2.4	Estimation de la DSP du bruit	73
5.2.5	Fonction de pondération modifiée	74

5.3 Implémentation de l'algorithme .....	75
5.4 Atténuation de l'écho résiduel .....	76
5.5 Résultats expérimentaux .....	78
5.5.1 Réduction de bruit dans le cas du simple duplex .....	78
5.5.2 Suppression d'écho résiduel dans le cas du simple duplex .....	82
5.5.3 Réduction du bruit et suppression de l'écho résiduel dans le cas du simple duplex .....	86
5.5.4 Réduction du bruit et suppression de l'écho résiduel dans le cas du full duplex .....	86
5.6 Conclusion .....	88
<b>Conclusion générale</b> .....	<b>89</b>
<b>Annexe A</b> .....	<b>90</b>
<b>Annexe B</b> .....	<b>95</b>
<b>Références bibliographiques</b>	



## Liste des Figures

1.1	Système de génération de l'écho acoustique	2
1.2	Annulation d'écho acoustique par filtrage adaptatif	4
1.3	Modélisation du système générateur de l'écho acoustique	5
1.4	Réponse impulsionnelle estimée et erreur de modélisation	6
1.5	Structure générale d'un modèle linéaire	7
1.6	Structure du modèle RIF	8
1.7	Structure du modèle ARX	9
1.8	Structure du modèle OE	10
1.9	Annulation d'écho stéréophonique	13
2.1	Configuration générale d'un filtre adaptatif	17
2.2	Identification dans le domaine temporel par filtrage adaptatif	20
2.3	Configuration générale d'un algorithme de filtrage adaptatif dans le domaine fréquentiel	26
2.4	Algorithme FBLMS	30
2.5	Convolution rapide et segmentation	33
2.6	Organigramme structurel de l'algorithme GMDF $\alpha$	37
2.7	Complexité de l'algorithme GMDF $\alpha$ par rapport au NLMS	40
3.1	Modèle de simulation et notations	42
3.2	Réponse impulsionnelle utilisée dans les simulations	44
3.3	Comparaison entre le NLMS et le GMDF	45
3.4	Comparaison entre le NLMS et le GMDF pour une entrée USASI	45
3.5	Influence du pas d'adaptation sur la convergence du GMDF contraint	46
3.6	Comparaison entre le GMDF contraint et le GMDF non contraint	46
3.7	Influence de la segmentation sur la convergence du GMDF contraint	47

3.8	Influence de la segmentation sur la convergence du GMDF non contraint	47
3.9	Influence du bruit additif sur la convergence du GMDF contraint	48
3.10	Annulation d'un écho acoustique non perturbé (Algorithme GMDF contraint)	50
3.11	Comparaison entre le NLMS et le GMDF pour un signal d'entrée non stationnaire	51
3.12	Annulation d'un écho acoustique non perturbé par un bruit additif	52
3.13	Annulation d'un écho acoustique non perturbé par un signal de parole	53
3.14	Annulation d'un écho acoustique non perturbé (Algorithme GMDF non contraint)	54
3.15	Effet d'une variation du système à identifier	55
4.1	Principe de Téléconférence avec annulation d'écho acoustique	57
4.2	Annulation d'écho acoustique dans le cas "Simple duplex" (Algorithme GMDF contraint)	60
4.3	Annulation d'écho acoustique dans le cas "Simple duplex" (Réponse impulsionnelle estimée)	61
4.4	Influence de la longueur de la réponse impulsionnelle sur l'annulation d'écho acoustique	62
4.5	Annulation d'écho acoustique dans le cas "Simple duplex" (Algorithme GMDF non contraint)	63
4.6	Annulation d'écho acoustique dans le cas "Full duplex"	66
4.7	Annulation d'écho acoustique avec détection du "Simple duplex"	67
4.8	Annulation d'écho acoustique dans le cas "Simple duplex" en présence du bruit	68
5.1	Diagramme bloc du système réducteur de bruit	75
5.2	Application de la soustraction spectrale à la réduction du bruit et la suppression d'écho résiduel	77
5.3	Réduction du bruit blanc Gaussien de moyenne nulle et de variance $\sigma^2 = 0.003$	79
5.4	Réduction du bruit dans une automobile (RSB = 10dB)	80
5.5	Réduction du bruit de climatisation dans un bureau (RSB = 20 dB)	81
5.6	Situation de simple duplexe avec suppression d'écho résiduel (RSB = 30 dB)	83
5.7	Situation de simple duplexe avec réduction de bruit d'automobile et suppression d'écho résiduel (RSB = 10 dB)	84
5.8	ERLE moyen avant et après réduction de bruit et suppression d'écho résiduel dans le cas simple duplex	85
5.9	Situation full duplex avec détection de double activité vocale (RSB = 10 dB)	87

## Introduction générale

Le mode de fonctionnement "mains-libres" est une caractéristique importante de beaucoup de terminaux actuels tels que les téléphones et les systèmes de téléconférence. En général, la qualité de la parole provenant d'un tel terminal est dégradée par la présence du bruit ambiant ainsi que de l'écho acoustique.

Dans ce projet de fin d'études on se propose de mettre en œuvre un annuleur d'écho acoustique basé sur un algorithme de filtrage adaptatif dans le domaine fréquentiel. De plus, une procédure de réduction du bruit est introduite par application de la technique de soustraction spectrale.

L'annulation de l'écho acoustique nécessite l'identification adaptative d'un système (chemin de l'écho) modélisé par une très longue réponse impulsionnelle (de quelques centaines à quelques milliers de coefficients). Une telle longueur rend impossible l'utilisation des algorithmes de type "moindres carrés" (charge de calcul prohibitive) ou du type "gradient stochastique" (vitesse de convergence trop faible). De plus, à la non stationnarité des signaux d'entrée (parole) s'ajoute la variation imprévisible du système (problème de poursuite) ainsi que la contrainte du délai de traitement qui doit être maintenu aussi faible que possible.

Dans la première partie de ce rapport, nous introduisons le phénomène de l'écho acoustique, ensuite nous exposons la dérivation d'un algorithme d'identification adaptative dans le domaine fréquentiel afin d'envisager son application à l'annulation d'écho acoustique. Dans la deuxième partie, nous effectuons l'expérimentation dans plusieurs contextes, de ce fait une approche de détection de double activité vocale exploitant la dissimilarité spectrale sera présentée afin d'améliorer les performances de l'annuleur d'écho dans une situation de full duplex.

Dans la dernière partie, nous abordons le problème de réduction du bruit par l'application de la technique de soustraction spectrale basée sur la méthode des statistiques minimales ainsi que son extension à la suppression de l'écho résiduel.

# Notations

Dans ce qui suit :

1. les *minuscules* correspondent aux variables dans le domaine temporel, les MAJUSCULES à celles dans le domaine fréquentiel et les caractères **gras** pour les vecteurs et les matrices.
2. les exposants <sup>T</sup>, \* et <sup>H</sup> correspondent respectivement à la transposition, la conjugaison complexe et la transposition conjuguée.
3.  $W_M$  et  $W_M^{-1}$  sont des matrices de taille  $M \times M$  dont les éléments sont respectivement les coefficients de la transformée de Fourier discrète (TFD) et de la TFD inverse :

$$[W_M]_{nk} = \left\{ e^{-j \frac{2\pi}{M} nk} \right\}$$

$$[W_M^{-1}]_{nk} = \frac{1}{N} \left\{ e^{j \frac{2\pi}{M} nk} \right\}$$

4.  $O_{N \times M}$  et  $I_N$  représentent les matrices nulle et identité, dont les indices indiquent leur dimension.

# Chapitre 1

## Introduction à l'annulation de l'écho acoustique

A travers ce chapitre, on décrira quelques caractéristiques relatives au phénomène de l'écho acoustique, indispensables aux développements ultérieurs. Ainsi, le concept de mécanisme de couplage acoustique est introduit. Par la suite, on présentera la technique générale d'annulation de l'écho. Enfin, un bref exposé résumant les différentes approches algorithmiques possibles sera présenté.

### 1.1 Introduction

Aux débuts de l'ère des télécommunications, il y a environ un siècle de cela, les utilisateurs étaient dans l'obligation de faire usage de leurs deux mains, l'une leur servait à tenir un microphone près de leur bouche, alors que l'autre servait à immobiliser un haut-parleur près d'une oreille. Il n'a pas fallu attendre longtemps pour bénéficier d'une main libre, le microphone et le haut-parleur étaient alors assemblés dans un même combiné. Cependant, le concept "mains-libres" n'était pas encore atteint.

Pour le commun des utilisateurs, il est encore difficile de comprendre qu'il faut mobiliser tous les outils de traitement de signal disponibles de nos jours afin d'apporter une solution efficace au problème de la téléphonie "mains-libres" utilisée en téléconférence. Un grand nombre de publications et de rapports portant sur l'état de l'art dans ce domaine ont été publiés et continuellement mis à jours ces dernières années. Différents algorithmes adaptatifs pour l'annulation de l'écho et la réduction de bruit ont suscité un intérêt certain.

## 1.2 L'écho acoustique

L'écho est un phénomène dans lequel une réplique retardée d'un son ou d'un signal électrique émis par une source primaire est réfléchi vers cette dernière. Dans le cas particulier de la téléconférence, l'écho acoustique a pour origine les différentes réflexions du son émis par le haut-parleur sur les différents objets contenus dans un environnement clos, par exemple une pièce, et captées par le microphone. L'écho peut aussi résulter d'une rétroaction entre le haut-parleur et le microphone. Notons que le signal émis par le microphone est constitué par la superposition du signal de parole du locuteur proche B et de l'écho généré par le signal de parole du locuteur lointain A (figure 1.1). Les effets plus ou moins néfastes d'un écho dépendent du délai temporel entre l'onde incidente et l'onde réfléchi, de l'intensité de cette dernière, ainsi que du nombre des différents échos générés à travers la multitude de parcours possibles. Dans le cas où le délai temporel ne serait pas très important, l'écho acoustique peut être perçu comme une douce réverbération qui améliore la qualité artistique d'une performance musicale dans une salle de concert ou d'un chant religieux. En revanche, un écho qui se manifeste dans un délai de plusieurs dizaines de millisecondes après l'onde incidente est indésirable, voire très gênant. Le système d'annulation de l'écho est par conséquent nécessaire dans ce cas.

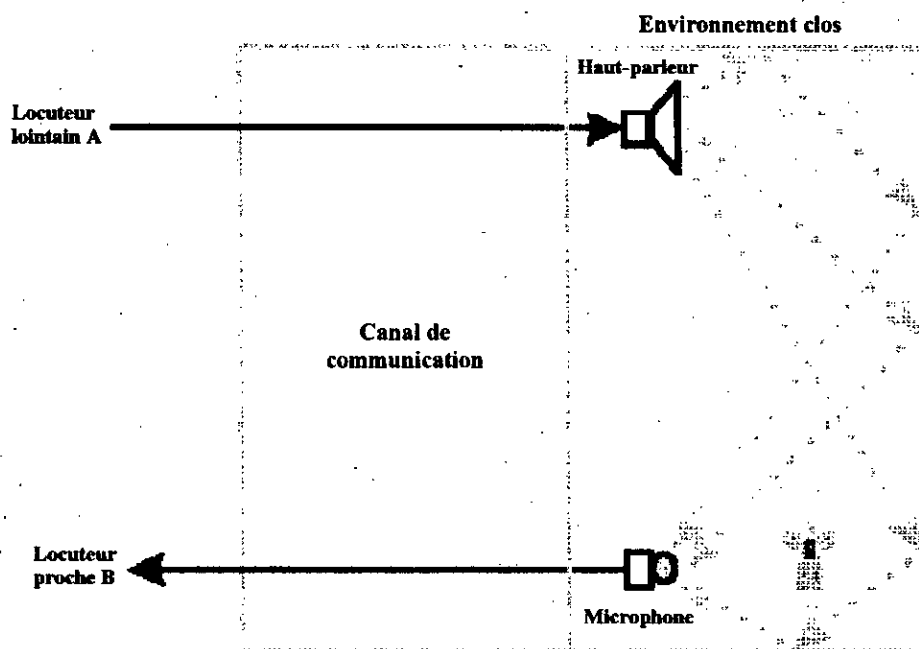


Figure 1.1 : Système de génération de l'écho acoustique

D'un point de vue pratique, le problème d'annulation de l'écho est assez compliqué pour plusieurs raisons :

- Le délai temporel de l'écho acoustique est très long (supérieur à une seconde). Ceci implique qu'un grand nombre de paramètres à estimer, probablement de l'ordre de quelques milliers, sont nécessaires dans un filtre RIF afin d'obtenir une approximation raisonnable.
- Les caractéristiques d'un chemin de l'écho sont plutôt non-stationnaires (ouverture ou fermeture d'une porte, mouvement d'une personne, etc.).
- L'écho acoustique résulte de la réflexion du son émis par le haut-parleur sur différentes surfaces (murs, fenêtres, plafonds, sols, etc.), et peut parcourir différentes trajectoires.
- La modélisation du chemin de l'écho par des filtres RIF ou RII linéaires n'est pas une bonne approximation à cause des nonlinéarités qui sont essentiellement dues à la caractéristique entrée-sortie du haut-parleur.

### 1.3 Mécanisme de couplage acoustique rétroactif

Le problème en téléphonie "mains-libres", (cas des systèmes de téléconférence ou des systèmes d'aide à l'audition), est le couplage rétroactif des ondes sonores entre les haut-parleurs et les microphones. La rétroaction acoustique peut être facilement mise en évidence si une portion significative de l'énergie sonore générée par le haut-parleur est réinjectée dans le microphone à travers une boucle de rétroaction. Si le gain total de la boucle, dépendant de la réponse fréquentielle du système ainsi excité, est supérieur à l'unité, le signal du locuteur lointain est alors amplifié dans une boucle fermée, ce qui aboutira à un hurlement rétroactif (Effet Larsen).

Plusieurs méthodes destinées à l'élimination d'un tel phénomène ont été proposées :

- Ajout d'un décalage en fréquence ou en phase dans la partie électrique du trajet bouclé, ainsi le signal subira une translation en fréquence de quelques hertz avant d'être retransmis par le haut-parleur. Cette méthode garantit une réduction de l'effet hurlement mais ne permet pas une élimination totale de l'écho.
- Réduction du gain du système bouclé aux fréquences pour lesquelles émergent des oscillations acoustiques en utilisant un filtre adaptatif. L'inconvénient d'une telle méthode est l'introduction d'une distorsion sur le signal utile.

- Utilisation d'un système adaptatif d'annulation de l'écho acoustique. Cette méthode représente la meilleure solution connue de nos jours pour la résolution de ce problème.

#### 1.4 Annulation de l'écho (AE)

Une méthode efficace destinée à l'élimination du signal de l'écho consiste à utiliser un annuleur de l'écho. Dans ce dernier on procède premièrement à l'estimation des caractéristiques (ou réponse impulsionnelle) du chemin de l'écho réel, par la suite une réplique du signal de l'écho réel est générée pour lui être soustraite. Dans la mesure où la réponse impulsionnelle du chemin de l'écho est inconnue a priori, une estimée de cette dernière doit être déterminée. Plusieurs méthodes qui seront discutées plus loin sont utilisées pour trouver l'approximation de la réponse impulsionnelle. Dans la mesure où cette dernière est variable dans le temps, une mise à jour adaptative est nécessaire. L'annulation de l'écho acoustique dans un environnement clos est présentée en figure 1.2 dans une situation de téléconférence.

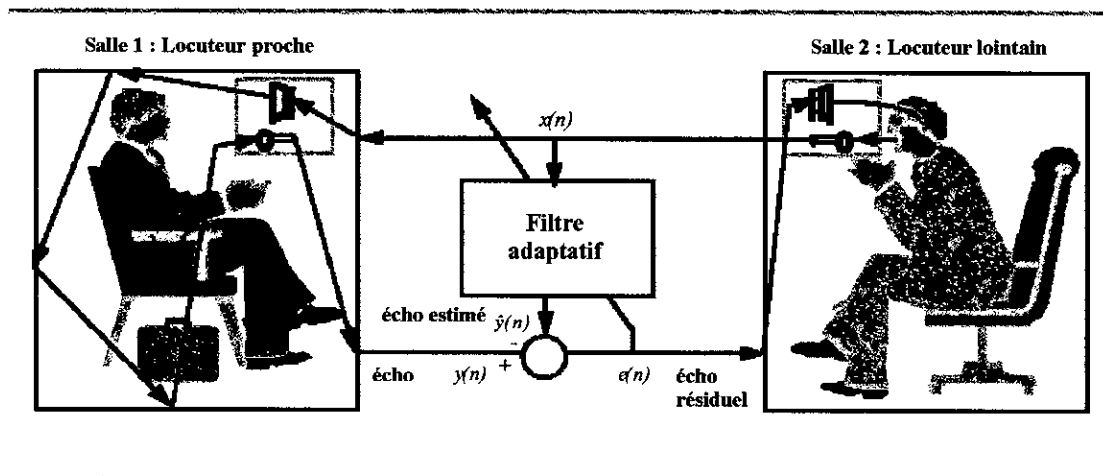


Figure 1.2 : Annulation de l'écho acoustique par filtrage adaptatif

L'écho estimé  $\hat{y}(n)$ , est généré par convolution du signal lointain  $x(n)$  provenant de la salle 2 avec le modèle du chemin de l'écho estimé par le filtre adaptatif. Soit  $h_e(n)$  la réponse impulsionnelle estimée modélisant le canal de l'écho réel, on exprime dans ce cas l'écho estimé par :  $\hat{y}(n) = h_e(n) * x(n)$  où  $*$  représente l'opérateur de convolution. Le signal d'erreur de l'annuleur de l'écho est alors obtenu par :  $e(n) = y(n) - \hat{y}(n)$ , ce dernier sera strictement nul si la modélisation est parfaite. Notons que le signal  $y(n)$  peut être vu comme étant le signal de l'écho réel ajouté au signal du locuteur proche pouvant être accompagné d'un bruit additif provenant de la salle 1 (perturbation inconnue). Ainsi, une bonne modélisation du canal de l'écho réel ne peut



être atteinte qu'en faisant abstraction de tout signal provenant de la première salle, on assumera ce dernier non corrélé avec le signal de l'écho dans ce qui suit.

## 1.5 Modélisations linéaires

Le système générateur de l'écho peut être approximativement décrit par un modèle linéaire. Avant d'entamer le problème d'estimation du chemin de l'écho acoustique, nous admettrons que ce dernier est linéaire et stationnaire sur de courts intervalles temporels. Bien que le système présente des caractéristiques nonlinéaires, l'hypothèse de linéarité donne des résultats acceptables dépendant principalement de la qualité de l'équipement électroacoustique utilisé, et en particulier du haut-parleur. Ainsi on voit qu'il est difficile d'assurer une perspicacité profonde au sujet du système générateur de l'écho pour chaque but spécifique. Par conséquent, il est préférable d'opter pour une approche modélisant le système par une boîte noire, en se basant sur l'observation des signaux d'entrée et de sortie notés respectivement  $x(n)$  et  $y(n)$ . Le but étant la modélisation du système inconnu (chemin de l'écho), on supposera que le signal de sortie mesuré est donné par l'écho  $r(n)$  additionné à la perturbation du locuteur proche  $u(n)$ , i.e.

$$y(n) = r(n) + u(n) \quad (1.1)$$

où  $r(n) = h(n) * u(n)$  et  $h(n)$  la réponse impulsionnelle du chemin de l'écho réel. Le système générateur de l'écho est schématisé en figure 1.3.

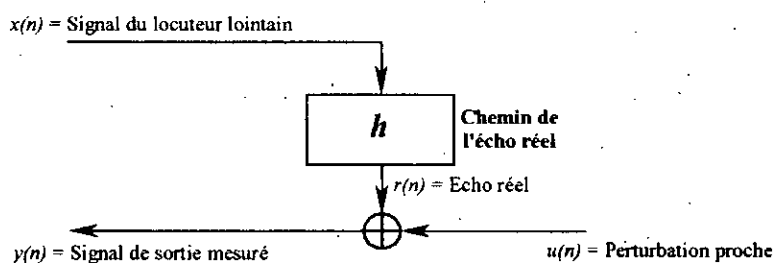


Figure 1.3 : Modélisation du système générateur de l'écho acoustique

Dans la mesure où le signal de l'écho réel est inconnu a priori, le but de la modélisation sera de reproduire le signal de l'écho à partir du signal d'entrée. Vu qu'il n'est pas possible d'obtenir le modèle exact du système réel  $h$ , il sera justifié d'introduire l'erreur de modélisation

$\varepsilon_m(n)$ , qui exprimera l'ajustement du modèle estimé  $h_e$  (voir figure 1.4). Ainsi, la modélisation sera parfaite pour une erreur de modélisation nulle.

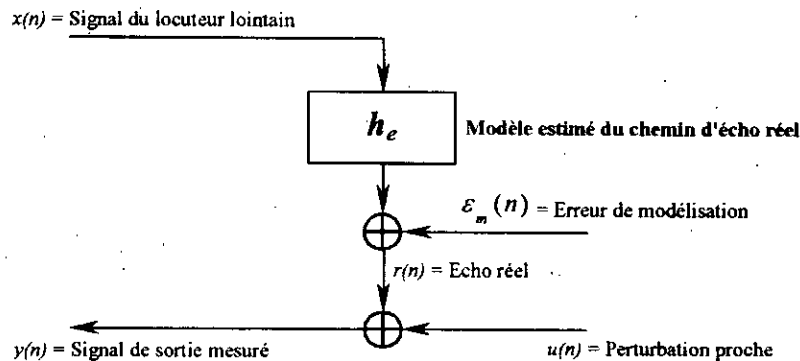


Figure 1.4 : Réponse impulsionnelle estimée et erreur de modélisation

En pratique, un système linéaire, stable et invariant dans le temps (LIT) peut être efficacement modélisé par une boîte noire en structure linéaire qui peut être exprimée par (voir figure 1.5) [12] :

$$A(q)y_0(n) = \frac{B(q)}{F(q)}x(n - nk) + \frac{C(q)}{D(q)}w(n) \quad (1.2)$$

où  $y_0(n)$ ,  $x(n)$  et  $nk$  représentent respectivement le signal de sortie optimal estimé, le signal d'entrée et le délai (en utilisant un modèle de classe spécifique et de taille fixée). Le bruit  $w(n)$  est une séquence blanche de moyenne nulle et de variance  $\sigma^2$ . Le décalage en temps est représenté par l'opérateur  $q$  vérifiant  $q^{-1}x(n) = x(n-1)$ . Chaque fonction de transfert polynomiale est donnée par :

$$\left. \begin{aligned} A(q) &= 1 + a_1q^{-1} + a_2q^{-2} + \dots + a_{na}q^{-na} \\ C(q) &= 1 + c_1q^{-1} + c_2q^{-2} + \dots + c_{nc}q^{-nc} \\ D(q) &= 1 + d_1q^{-1} + d_2q^{-2} + \dots + d_{nd}q^{-nd} \\ F(q) &= 1 + f_1q^{-1} + f_2q^{-2} + \dots + f_{nf}q^{-nf} \\ B(q) &= 1 + b_1q^{-1} + b_2q^{-2} + \dots + b_{nb}q^{-nb} \end{aligned} \right\} \quad (1.3)$$

où  $na$ ,  $nc$ ,  $nd$ ,  $nf$  et  $nb$  sont les ordres fixés de chaque polynôme. Plusieurs structures de modèles peuvent être dérivées de l'équation (1.2). Par exemple, le modèle RIF (Réponse Impulsionnelle Finie) est obtenu pour  $A(q)=C(q)=D(q)=F(q)=1$ , le modèle ARX (Auto Régressif avec entrée eXterne) correspond à  $C(q)=D(q)=F(q)=1$ , le modèle OE (Output Error)

est obtenu quand  $A(q)=C(q)=D(q)=1$ , etc. Notons que,  $A(q)$  représente les pôles communs au signal d'entrée et au bruit, et de façon similaire,  $F(q)$  représente les pôles caractérisant uniquement le signal d'entrée, alors que  $D(q)$  détermine les pôles correspondant uniquement au bruit. L'avantage d'un tel modèle réside dans le fait que la perturbation est également prise en considération et ce dans le but d'obtenir un résultat plus précis. En conséquence, dans la mesure où la perturbation proche (locuteur proche et bruit local) est inconnue a priori, il est plus convenant de modéliser  $u(n)$  par une source de bruit blanc filtré par un système linéaire  $[C(q)/D(q)A(q)]$ . L'erreur de modélisation  $\varepsilon_m(n)$  représentera les défauts de modélisation de la relation existant entre le signal d'entrée et le signal de sortie.

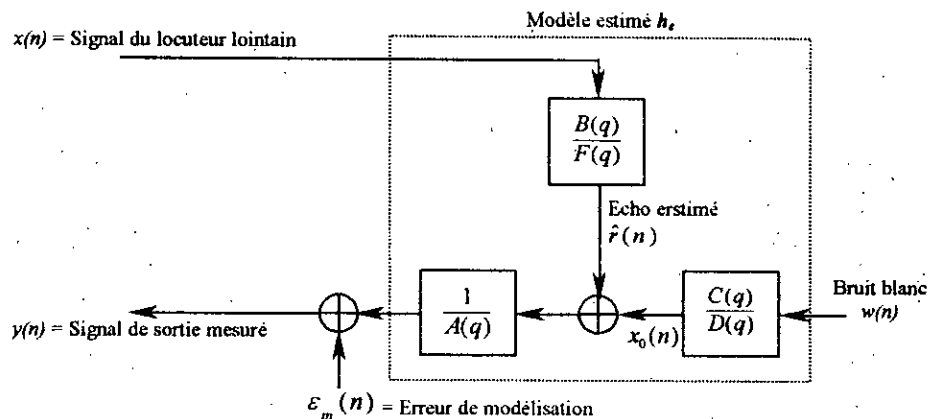


Figure 1.5 : Structure générale d'un modèle linéaire

### 1.5.1 Structure du modèle à Réponse Impulsionnelle Finie (RIF)

Le modèle RIF est largement utilisé, ceci est essentiellement dû à la simplicité de sa structure ainsi qu'à sa stabilité. Son expression étant un cas particulier de l'équation (1.2) qui est obtenue en posant  $A(q)=C(q)=D(q)=F(q)=1$  :

$$\begin{aligned}
 y_0(n) &= B(q)x(n - nk) + w(n) \\
 &= \sum_{k=0}^{nb} b_k q^{-k} x(n - nk) + w(n) \\
 &= (1 + b_1 q^{-1} + b_2 q^{-2} + \dots + b_{nb} q^{-nb})x(n - nk) + w(n) \\
 &= x(n - nk) + b_1 x(n - nk - 1) + b_2 x(n - nk - 2) + \dots + b_{nb} x(n - nk - nb) + w(n) \quad (1.4)
 \end{aligned}$$

où les paramètres  $b_k$ , pour  $k=0, 1, \dots, nb$  représentent les coefficients de la réponse impulsionnelle du système. La figure 1.6 illustre le modèle RIF. Observons que, le signal  $w(n)$  peut être perçu

comme le signal de parole du locuteur proche et/ou la perturbation inconnue (bruit additif éventuel).

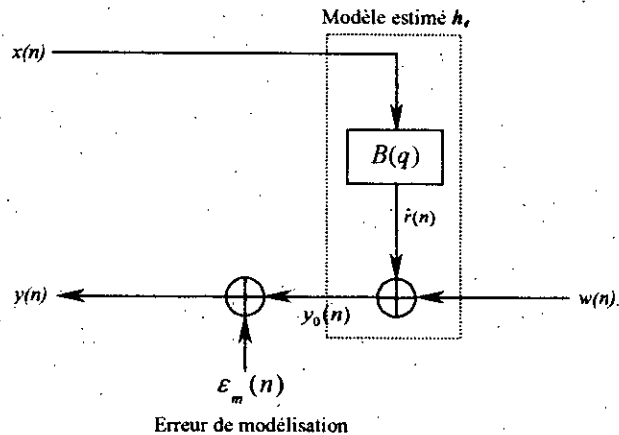


Figure 1.6 : Structure du modèle RIF

Evidemment, le modèle RIF est bien approprié pour représenter le chemin de l'écho acoustique dans la mesure où il ne modélise pas le signal  $w(n)$ , ce dernier étant considéré comme le signal utile dans notre cas. Cependant, cette structure nécessite un grand nombre de paramètres à estimer dans le but d'obtenir une approximation précise. Théoriquement, le nombre de paramètres employés dans l'opération de filtrage dépend du délai de parcours du chemin de l'écho par l'onde acoustique d'une part, et de la taille de la réponse impulsionnelle modélisant le chemin de l'écho d'autre part. Afin d'obtenir de bonnes performances, le nombre de paramètres doit être supérieur aux effets combinés de la taille de la réponse impulsionnelle du chemin de l'écho réel et du délais de parcours de ce dernier. Soit dans ce cas  $T_s$  la période d'échantillonnage du signal de parole numérisé,  $L$  le nombre de paramètres utilisés dans l'opération de filtrage, et  $\tau$  la durée des effets combinés à considérer. On obtient [12] :

$$LT_s > \tau \quad (1.5)$$

Partant du fait qu'une valeur typique de  $T_s$  est de 125  $\mu$ s pour un système de communication numérique standard, il est évident que, si  $\tau = 500$  ms,  $L > 4000$  paramètres sont requis dans le filtre.

### 1.5.2 Structure du modèle Auto Régressif avec entrée eXterne ARX

La structure de ce modèle peut être extraite de l'équation (1.2) en posant  $C(q)=D(q)=F(q)=1$ . Dans ce cas, les pôles représentés par  $A(q)$  peuvent être éventuellement utilisés pour modéliser le système avec un nombre moindre de paramètres [10]. Le signal de

chemin de l'écho. Ceci se justifie par le fait qu'il est possible d'exprimer la fonction de transfert indépendamment de la perturbation locale, et que le nombre de paramètres à estimer s'en trouverait nettement réduit par rapport au modèle RIF [12].

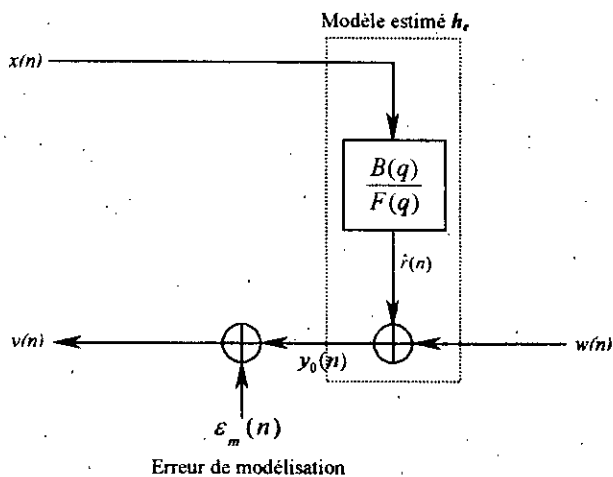


Figure 1.8 : Structure du modèle OE

Ce modèle peut être déduit de l'équation (1.2) en posant  $A(q)=C(q)=D(q)=1$ . Il s'exprime comme suit :

$$\begin{aligned} y_0(n) &= \hat{r}(n) + w(n) \\ &= \frac{B(q)}{F(q)} x(n - nk) + w(n) \end{aligned} \quad (1.7)$$

ou, également,

$$\sum_{k=0}^{nf} f_k y_0(n-k) = \sum_{k=1}^{nb} b_k x(n-nk-k) + \sum_{k=0}^{nf} f_k w(n-k) \quad (1.8)$$

Dans la mesure où  $f_0=1$ , on obtient :

$$[y_0(n) - w(n)] = \sum_{k=1}^{nb} b_k x(n-nk-k) - \sum_{k=1}^{nf} f_k [y_0(n-k) - w(n-k)] \quad (1.9)$$

Et prenant en considération le fait que  $\hat{r}(n) = y_0(n) - w(n)$ , l'équation précédente peut être réécrite comme une régression des entrées mesurées et des sorties estimées :

$$\hat{r}(n) = \sum_{k=1}^{nb} b_k x(n-nk-k) - \sum_{k=1}^{nf} f_k \hat{r}(n-k) \quad (1.10)$$

Dans ce cas, le vecteur des paramètres à estimer est  $[b_1 \ b_2 \ \dots \ b_{nb} \ f_1 \ f_2 \ \dots \ f_{nf}]^T$ . Et on voit clairement que le modèle OE peut représenter efficacement le chemin de l'écho dans la mesure où il permet une réduction du nombre de paramètres à estimer, et ce sans modéliser pour autant

le signal  $w(n)$ . Néanmoins, on peut subir quelques difficultés en utilisant le modèle OE. Premièrement, on ne peut garantir absolument la stabilité du système. Deuxièmement, la présence de l'ancienne sortie prédite dans l'équation (1.10) empêche l'utilisation d'une simple solution linéaire pour trouver les paramètres  $b_k$  et  $f_k$  [12]. Par conséquent, nous ne considérerons pas ce modèle dans ce qui suit.

## 1.6 Structures algorithmiques

Plusieurs types d'algorithmes adaptatifs ont été appliqués à l'annulation de l'écho acoustique, on présente dans ce qui suit les différentes approches algorithmiques possibles, ainsi que les différents avantages et inconvénients relatifs à chacune d'elles.

### 1.6.1 Structures dérivées de l'algorithme du gradient

Le NLMS (Normalised Least Mean Square) est l'algorithme de cette famille qui est le plus facilement et le plus fréquemment implémenté. Il se base sur la minimisation de l'erreur quadratique moyenne. La complexité arithmétique du NLMS est relativement faible comparée à celle exigée par les autres algorithmes. Ceci est important dans la mesure où l'application considérée (annulation de l'écho) exige un grand nombre de coefficients du filtre. L'inconvénient est une convergence lente, en particulier pour des signaux d'entrée fortement corrélés. L'introduction du traitement par blocs [16], ainsi que la mise à profit des méthodes de convolution rapides dans le domaine fréquentiel [17] (voir annexe B), permet de palier à ce problème. De plus, l'ajout d'un degré de liberté supplémentaire qu'est le facteur de suréchantillonnage  $\alpha$  [1], conduit à une mise à jour plus fréquente du filtre adaptatif et améliore ainsi la vitesse de convergence et la capacité de poursuite. Cependant, les données étant découpées en blocs de taille liée à la longueur de la réponse impulsionnelle, la causalité du filtrage entraîne alors des délais prohibitifs inacceptables dans des applications en temps réel. Une segmentation de la réponse impulsionnelle en petits blocs permet de palier à cette limitation structurelle [28].

### 1.6.2 Algorithmes des moindres carrés récursifs

L'algorithme RLS (Recursive Least Squares) est connu pour sa rapidité de convergence. Cependant si l'algorithme est implémenté en précision finie, il peut présenter des instabilités numériques pour des erreurs d'approximation croissantes. Si l'on doit faire face à un grand

nombre de coefficients du filtre, une implémentation directe de l'algorithme RLS n'est pas faisable dans la mesure où sa complexité arithmétique est de l'ordre de  $L^2$  où  $L$  est la taille du filtre. Plusieurs approches aboutissant à des versions rapides de l'algorithme RLS sont connues pour être principalement basées sur des techniques de pré-fenêtrage, ces dernières permettent une réduction de la charge de calcul à une valeur de l'ordre de  $L$ , aboutissant à une implémentation rapide de l'algorithme RLS appelée FTF (Fast Transversal Filter). Malheureusement, l'algorithme FTF présente également des instabilités numériques et tend à diverger. Une idée de base réside en une extension de l'algorithme par accumulation des erreurs d'approximation et l'exécution de corrections appropriées dès que les erreurs numériques atteignent un certain seuil significatif [8].

### 1.6.3 Algorithme de Newton rapide

L'algorithme de Newton rapide peut être vu comme une version simplifiée de l'algorithme RLS. Dans le cas des implémentations rapides de ce dernier, des prédictions linéaires d'ordre  $L$  sont nécessaires, alors que dans le cas où l'ordre de corrélation du signal d'excitation serait faible, il n'est plus indispensable de calculer entièrement le vecteur de prédiction d'ordre  $L$ . Diminuant ainsi la taille du vecteur de prédiction à un ordre  $P$  approprié au signal d'excitation, on aboutit à l'algorithme de Newton rapide. La performance ainsi atteinte du point de vue convergence est comparable à celle de l'algorithme RLS, alors que la complexité est de l'ordre de  $LP$  [8].

### 1.6.4 Algorithmes de filtrage adaptatif en sous-bandes

Dans le cas du filtrage en sous-bandes, les signaux d'entrée et de sortie du système sont décomposés en plusieurs bandes fréquentielles adjacentes. A chaque signal de sous-bande sous-échantillonné est alors appliqué l'algorithme d'identification. Le résidu est par la suite synthétisé par interpolation au moyen de filtres de synthèse (pouvant être reliés aux filtres de décomposition par des relations de reconstruction parfaite [11]) puis sommation des signaux sous-bande.

Le traitement de signaux sous-échantillonnés réduit la charge de calcul. D'autre part l'identification sur les signaux sous-bande améliore la vitesse de convergence des algorithmes de descente du gradient, car la dynamique spectrale des signaux dans les sous-bandes est significativement plus faible que la dynamique spectrale du signal pleine bande. Cependant le sous-échantillonnage peut faire apparaître des versions image de l'entrée dans la sortie. Parmi les méthodes proposées pour résoudre ce problème, on citera :

- Le non recouvrement des filtres adjacents créant des sauts de plages de fréquence entre les sous-bandes (nécessité d'un nombre élevé de sous-bandes).
- L'utilisation de filtres croisés entre sous-bandes (difficulté d'adaptation de ces filtres croisés) [11].

## 1.7 Annulation de l'écho stéréophonique

Récemment, l'annulation de l'écho stéréophonique est devenue de plus en plus répandue pour des applications de téléconférence ou de jeux vidéo [8]. Comme les signaux d'excitation sont corrélés (voir figure 1.9), il n'existe pas de solution unique pour l'identification des deux réponses impulsionnelles. De plus, une matrice de corrélation étendue des deux signaux d'entrée doit être inversée [13]. Dans le cas d'une forte corrélation, des instabilités numériques dues à un mauvais conditionnement auront pour conséquence une divergence des paramètres des deux filtres. Cependant, il existe un certain nombre d'approches pour surmonter la corrélation des deux signaux d'excitation. Une technique applique une fonction de non linéarité à l'un des deux signaux d'excitation [13]. Dans une seconde approche, la matrice de corrélation est régularisée par introduction d'un facteur de fuite dans la mise à jour du vecteur des paramètres des filtres.

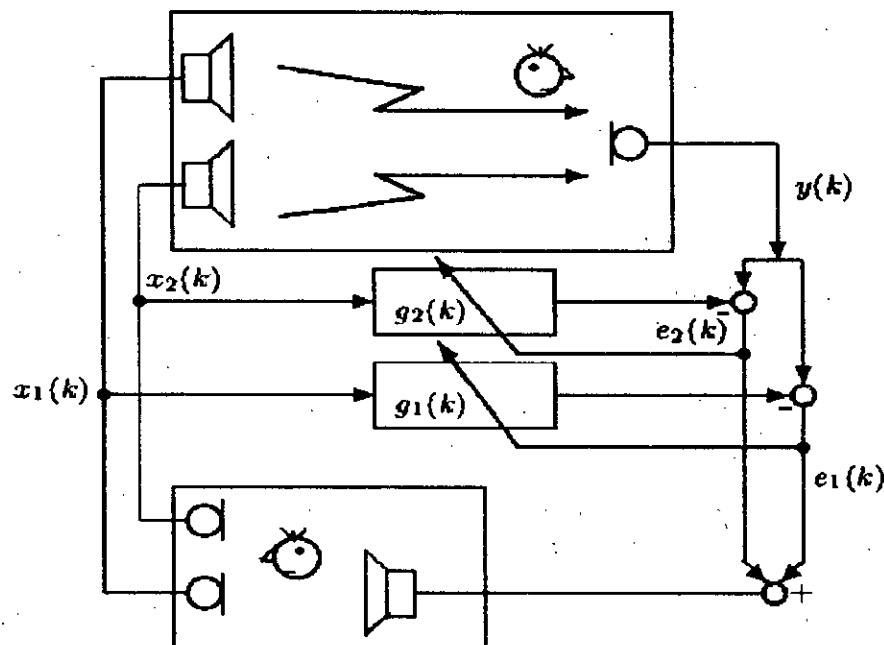


Figure 1.9 : Annulation de l'écho stéréophonique

## 1.8 Conclusion

Les systèmes de téléphonie dotés d'équipements de communication "mains-libres" existent sur le marché. Les possibilités de communication en full duplex ainsi que la réduction



des effets du bruit ambiant sont offertes avec la garantie d'un certain niveau de performances. Cependant, la nécessité d'améliorer les performances peut être comblée par une meilleure compréhension du problème ainsi que par la disponibilité de processeurs de traitement du signal encore plus puissants. Aussi longtemps que la compréhension plus approfondie du problème sera concernée, l'amélioration des différents algorithmes comprenant une procédure d'annulation de l'écho acoustique et de réduction de bruit sera la voie la plus prometteuse. Donc, même après plus d'une décennie de recherche et de développement, le défi est toujours existant.

Ce bref chapitre nous a permis de mettre en évidence la nécessité de recourir à une modélisation du chemin de l'écho acoustique, ainsi le choix portant sur le modèle RIF est justifié par deux raisons principales. La première est liée à la stabilité de ce modèle pendant l'exécution du processus d'adaptation [7]. Un deuxième avantage important se manifeste par l'absence de minimums locaux dans le cas de l'utilisation d'une *fonction objective* quadratique, autrement dit le modèle présente une solution optimale unique [3].

On aborde dans le chapitre suivant une étude détaillée des algorithmes de filtrage adaptatif dérivés de l'algorithme du gradient dans le domaine temporel et fréquentiel. Ainsi les différentes étapes permettant d'aboutir à une structure générale globalisant les avantages relatifs à chaque structure particulière seront abordées. On justifie par ailleurs ce choix par les raisons précédemment citées, à savoir la mise à profit des avantages de stabilité numérique ainsi que la faible complexité mise en jeu.

## Chapitre 2

# Algorithmes de filtrage adaptatif

Le traitement adaptatif du signal est un outil très important utilisé dans plusieurs applications. Il s'agit d'une classe d'algorithmes d'optimisation mis en œuvre en temps réel et assurant automatiquement la mise à jour d'un système en fonction de son environnement et permettant une adaptation optimale lors des fluctuations de ce dernier. En pratique ces propriétés sont fondamentales car elles évitent l'intervention d'un opérateur et permettent un fonctionnement sans coupure au cours du temps.

Ce qui nous intéresse dans cette étude est de rechercher l'algorithme de filtrage adaptatif offrant les meilleures caractéristiques, en terme de complexité arithmétique et vitesse de convergence, afin d'envisager son application à l'annulation d'écho acoustique où le chemin d'écho doit être modélisé par une très longue réponse impulsionnelle (RIF) pour obtenir un niveau de performances satisfaisant.

Dans ce chapitre, nous allons tout d'abord présenter quelques notions générales sur le filtrage adaptatif, puis nous décrivons les algorithmes temporels et fréquentiels. L'évolution de ces algorithmes conduit à la dérivation de l'algorithme  $\text{GMDF}\alpha$  que nous détaillerons à la fin de ce chapitre.

## 2.1 Introduction au filtrage adaptatif

Lors de la conception d'un filtre invariant dans le temps, les paramètres internes ainsi que la structure globale du filtre sont fixes, et si de plus le filtre est linéaire, le signal de sortie sera une combinaison linéaire de la séquence d'entrée, un tel filtre est un système linéaire invariant dans le temps (LIT), dont la conception se base sur la spécification au préalable de ses caractéristiques fixes. Cependant, dans le cas où ces spécifications seraient inconnues a priori, ou que ces dernières ne peuvent être satisfaites par un filtre invariant dans le temps, un filtrage adaptatif est nécessaire. Strictement parlé, un filtre adaptatif est considéré comme un filtre nonlinéaire dans la mesure où ses caractéristiques dépendent du signal d'entrée, et par conséquent, les conditions d'homogénéité et d'additivité ne sont pas satisfaites. Cependant, si on fige les paramètres du filtre à un instant donné, on peut considérer, sous cette condition, que le filtre adaptatif est linéaire au sens que le signal de sortie est une fonction linéaire du signal d'entrée.

Les filtres adaptatifs sont des systèmes variants dans le temps, dans la mesure où leurs caractéristiques changent en continu dans le but d'atteindre certaines performances exigées. Usuellement, la définition du critère de performance requiert l'existence d'un signal d'entrée et d'un signal de référence caractérisants l'environnement dans le but de concevoir, au mieux, le filtre adapté avec les exigences préalablement définies. La solution apportée par le filtrage adaptatif consiste en l'exécution d'une mise à jour des paramètres du filtre par l'intermédiaire d'un algorithme exploitant uniquement les informations disponibles sur l'état de l'environnement.

### 2.1.1 Structure générale d'un filtre adaptatif

La disposition générale d'un environnement de filtrage adaptatif est illustrée par la figure 2.1, où  $x(n)$  est le signal d'entrée,  $y(n)$  est le signal de sortie estimée,  $d(n)$  le signal de sortie désirée (la quantité que l'on veut reproduire) et  $e(n)$  le signal d'erreur défini comme étant la différence entre la sortie désirée et la sortie estimée. Ce dernier est utilisé afin d'établir *la fonction objective* requise par l'algorithme d'adaptation dans le but de déterminer la mise à jour appropriée des coefficients du filtre.

Comme montré dans la figure 2.1, la conception d'un filtre adaptatif impose la caractérisation de deux structures principales : la structure de filtrage et la structure d'adaptation.

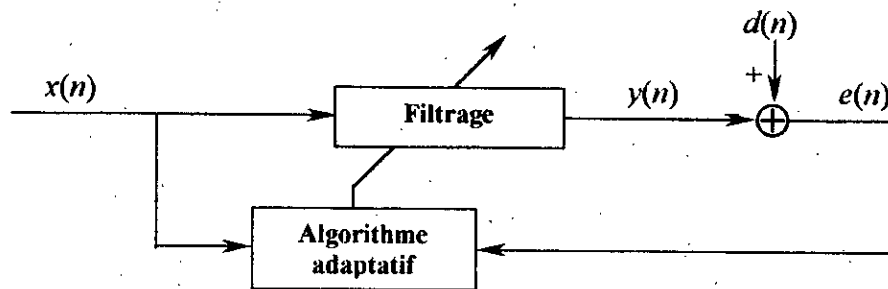


Figure 2.1 : Configuration générale d'un filtre adaptatif

- **la structure de filtrage**

La conception du filtre adaptatif impose le choix d'une structure de filtrage appropriée. Cette dernière dépend des caractéristiques du système à identifier et du nombre de modélisations possibles, on prendra en considération l'influence de la complexité de calcul (nombre d'opérations arithmétiques par itération) du processus ainsi que le nombre d'itérations nécessaires afin d'atteindre un certain niveau de performances.

- **La structure d'adaptation**

Cette structure définit la procédure à utiliser pour la mise à jour du vecteur des paramètres  $\theta(k)$  du filtre. La conception de cette partie du filtre impose deux étapes préliminaires :

- Choix de la fonction objective  $F\{e(n)\}$**

La fonction objective  $F\{e(n)\}$  pouvant être définie sous différentes formes, doit néanmoins, dans tous les cas de figure, satisfaire les propriétés d'optimisation et de non négativité. En tenant compte de la complexité de calcul de l'algorithme comme critère, on peut citer les fonctions objectives suivantes les plus utilisées dans l'établissement des algorithmes de filtrage adaptatif [6] :

- l'erreur quadratique instantanée (EQI) :

$$F\{e(n)\} = |e(n)|^2 \quad (2.1)$$

- les moindres carrés récursifs (MCR) :

$$F\{e(n)\} = \frac{1}{n+1} \sum_{k=0}^n |e(n-k)|^2 \quad (2.2)$$

- les moindres carrés récurrents pondérés (MCRP) :

$$F\{e(n)\} = \sum_{k=0}^n \lambda^{n-k} |e(n-k)|^2 \quad (2.3)$$

où  $\lambda$  est une constante appelée *facteur d'oubli*, dont la valeur se situe dans l'intervalle  $[0,1]$ .

- l'erreur quadratique moyenne (EQM) :

$$F\{e(n)\} = E\{|e(n)|^2\} \quad (2.4)$$

où l'opérateur  $E\{\bullet\}$  indique l'espérance mathématique.

## b. Définition de l'algorithme de minimisation de la fonction objective

L'algorithme de minimisation est défini comme étant la procédure à utiliser pour ajuster les paramètres du filtre adaptatif afin de minimiser la fonction objective. L'algorithme de minimisation affecte essentiellement la vitesse de convergence et la complexité de calcul du processus adaptatif. Les méthodes d'optimisation les plus communément utilisées dans le domaine de traitement du signal sont [6] :

- **La méthode de Newton**

Cette méthode recherche le minimum d'une approximation du second ordre de la fonction objective en utilisant une formule de mise à jour itérative du vecteur exprimant les coefficients du filtre :

$$\theta(n+1) = \theta(n) - \mu H_{\theta}^{-1} [F\{e(n)\}] \nabla_{\theta} [F\{e(n)\}] \quad (2.5)$$

où  $\mu$  est le pas d'adaptation de l'algorithme, i.e., un facteur déterminant la vitesse de variation du vecteur des coefficients du filtre. Et  $H_{\theta} [F\{e(n)\}]$  la matrice de taille  $L \times L$  dont les éléments sont les dérivées partielles de second ordre de la fonction objective par rapport aux coefficients du filtre :

$$[H_{\theta} [F\{e(n)\}]]_{kl} = \frac{\partial^2 F\{e(n)\}}{\partial \theta_k \partial \theta_l} \quad (2.6)$$

où  $\theta_k$  et  $\theta_l$  sont respectivement le  $k^{ème}$  et  $l^{ème}$  coefficient du filtre.

$\nabla_{\theta} [F\{e(n)\}]$  est le vecteur du gradient de la fonction objective par rapport aux coefficients du filtre :

$$[\nabla_{\theta} [F\{e(n)\}]]_k = \frac{\partial F\{e(n)\}}{\partial \theta_k} \quad (2.7)$$

- **Les méthodes quasi-Newton**

Cette classe d'algorithmes, qui sont des versions simplifiées de la méthode de Newton, tendent à minimiser la fonction objective en calculant de manière récursive une estimée de la matrice  $H_{\theta}^{-1} [F\{e(n)\}]$ , i.e. :

$$\theta(n+1) = \theta(n) - \mu P(n) \nabla_{\theta} [F\{e(n)\}] \quad (2.8)$$

où  $P(n)$  est une estimée de  $H_{\theta}^{-1} [F\{e(n)\}]$ , vérifiant :

$$\lim_{n \rightarrow \infty} P(n) = H_{\theta}^{-1} [F\{e(n)\}]$$

L'une des estimations usuelles de  $H_{\theta}^{-1} [F\{e(n)\}]$  est l'utilisation du lemme d'inversion matricielle [5]. D'autre part le vecteur gradient est généralement remplacé par une estimée, dont le but est un gain en charge de calcul.

- **La méthode du gradient stochastique**

Ce type d'algorithmes se base sur la recherche du point pour lequel la fonction objective est minimale en suivant la direction opposée au vecteur du gradient de cette fonction (voir annexe A). Par conséquent, l'équation de mise à jour prend la forme suivante :

$$\theta(n+1) = \theta(n) - \mu \nabla_{\theta} [F\{e(n)\}] \quad (2.9)$$

Dans le cas général, les méthodes du gradient sont faciles à implémenter mais leur vitesse de convergence est très faible, d'autre part, la méthode de Newton requiert usuellement un nombre moindre d'itérations pour atteindre le voisinage du point minimal de la fonction objective. Dans la plupart des cas, les méthodes quasi-Newton sont considérées comme des solutions optimales assurant un bon compromis entre charge de calcul et vitesse de convergence, néanmoins, on leur reprochera leurs problèmes d'instabilité, dont la cause est la forme récursive utilisée lors de l'estimation de la matrice  $H_{\theta}^{-1} [F\{e(n)\}]$ .

On notera que quelque soit la méthode de minimisation à adopter, le pas d'adaptation  $\mu$  sert à contrôler la stabilité, la vitesse de convergence, ainsi qu'un certain nombre de caractéristiques liées à l'erreur résiduelle du processus adaptatif (voir annexe A). Ainsi le choix du pas d'adaptation dépend de l'application pour laquelle l'algorithme adaptatif est destiné et les performances exigées. En pratique, une valeur appropriée de ce paramètre découlera d'un certain nombre d'essais, qui fourniront le meilleur outil pour surmonter ce problème.

Dans la suite de ce travail nous allons nous étaler sur les algorithmes dérivés de l'algorithme gradient se basant sur la minimisation de l'erreur quadratique moyenne.

### 2.2 Algorithmes de filtrage adaptatif dans le domaine temporel

Considérons l'expérience d'identification représentée dans la figure 2.2 où un filtre adaptatif, possédant  $L$  coefficients, est excité par un signal d'entrée  $x_n$ . Les coefficients de ce filtre sont adaptés itérativement de façon à minimiser la moyenne quadratique de l'erreur  $e_n$ , définie comme étant la différence entre la sortie actuelle  $y_n$  du filtre et le signal désiré  $d_n$ .

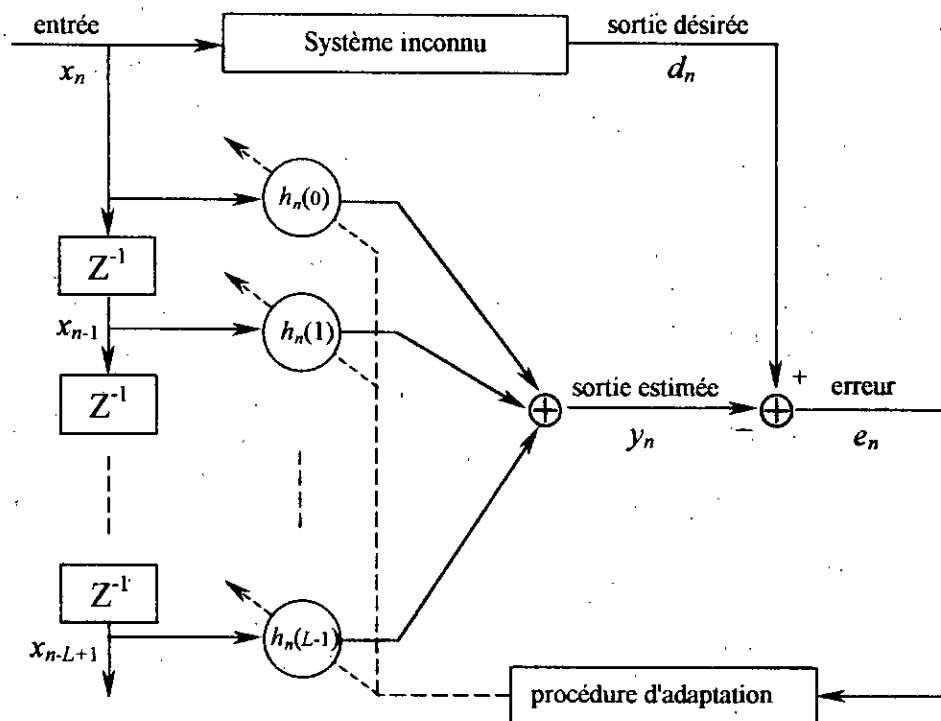


Figure 2.2 : Identification dans le domaine temporel par filtrage adaptatif

Une des procédures directes utilisée pour résoudre ce problème est l'algorithme de filtrage adaptatif dit LMS (*Least Mean Square Algorithm*) proposé initialement par *Widrow et Hoff* [15].

#### 2.2.1 Algorithme LMS

Cet algorithme est un cas particulier de l'algorithme du gradient stochastique se basant sur la minimisation de l'erreur quadratique moyenne définie par :

$$\xi(n) = E\{|e(n)|^2\} \tag{2.10}$$

La sortie du filtre adaptatif s'exprime par une convolution entre les coefficients actuels du filtre et la séquence d'entrée :

$$y(n) = \sum_{k=0}^{L-1} h(k) x(n-k) = \mathbf{h}_n^T \mathbf{x}_n = \mathbf{x}_n^T \mathbf{h}_n \quad (2.11)$$

où  $\mathbf{h}_n$  est le vecteur des coefficients du filtre à l'instant  $n$  :

$$\mathbf{h}_n = [h_n(0), \dots, h_n(L-1)]^T \quad (2.12)$$

et  $\mathbf{x}_n$  le vecteur des échantillons nécessaires pour le calcul de la sortie du filtre à l'instant  $n$  :

$$\mathbf{x}_n = [x(n), \dots, x(n-L+1)]^T \quad (2.13)$$

L'erreur est définie comme étant la différence entre la sortie du filtre adaptatif et la sortie désirée (la quantité que l'on veut reproduire) :

$$e(n) = d(n) - y(n) \quad (2.14)$$

En utilisant la méthode du gradient pour la recherche des coefficients du filtre minimisant l'erreur quadratique moyenne, l'équation de mise à jour s'exprime par :

$$\mathbf{h}_{n+1} = \mathbf{h}_n - \mu \nabla_{\mathbf{h}} \{\xi(n)\} \quad (2.15)$$

où  $\mu$  est le pas d'adaptation contrôlant la convergence et la stabilité de l'algorithme.

Le gradient  $\nabla_{\mathbf{h}} \{\xi(n)\}$  est défini comme étant la dérivée de l'erreur quadratique moyenne par rapport au coefficients du filtre [6] :

$$\nabla_{\mathbf{h}} \{\xi(n)\} = \frac{\partial \xi(n)}{\partial \mathbf{h}_n} = - E\{e(n) \mathbf{x}^*(n)\} \quad (2.16)$$

La mise à jour s'effectue alors suivant :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu E\{e(n) \mathbf{x}^*(n)\} \quad (2.17)$$

Dans les situations pratiques, l'espérance  $E\{e(n) \mathbf{x}^*(n)\}$  ne peut pas être évaluée. Cependant, celle-ci peut être approchée par une estimation en utilisant la moyenne échantillonnée sur  $M$  observations :

$$\hat{E}\{e(n) \mathbf{x}^*(n)\} = \frac{1}{M} \sum_{k=0}^{M-1} e(n-k) \mathbf{x}^*(n-k) \quad (2.18)$$

L'équation d'adaptation devient alors :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu \frac{1}{M} \sum_{k=0}^{M-1} e(n-k) \mathbf{x}^*(n-k) \quad (2.19)$$



Dans le cas particulier où  $M=1$ , l'équation de mise à jour s'exprime par :

$$h_{n+1} = h_n + \mu e(n) x^*(n) \quad (2.20)$$

Cette équation définit l'algorithme LMS [15].

La simplicité de cet algorithme réside dans le fait que l'adaptation de chaque coefficient du filtre ne nécessite qu'une seule addition et une seule multiplication (la quantité  $\mu e(n)$  est calculée une seule fois pour tous les coefficients du filtre).

D'autre part, le choix du pas d'adaptation  $\mu$  doit assurer la convergence de l'algorithme vers la solution optimale. Ceci est possible en imposant (voir annexe A) :

$$0 < \mu < \frac{2}{\lambda_{max}} \quad ; \text{ pour une convergence en moyenne.}$$

$$0 < \mu < \frac{2}{tr(R_x)} \quad ; \text{ pour une convergence en moyenne quadratique.}$$

où  $\lambda_{max}$  est la valeur propre maximale de la matrice d'auto-corrélation  $R_x$  du signal d'entrée et  $tr(R_x)$  est la trace de  $R_x$ , i.e. la somme de ses éléments diagonaux.

Dans les situations pratiques l'algorithme LMS présente deux problèmes : d'une part, le choix du pas d'adaptation assurant la convergence de l'algorithme dépend des caractéristiques du signal d'entrée, qui en pratique sont inconnues et variables. D'autre part le terme correctif qui est appliqué aux coefficients du filtre est proportionnel au vecteur d'entrée, ce qui peut poser un problème d'amplification du bruit du gradient pour de grandes amplitudes du signal d'entrée [3 – 6]. Ces deux problèmes ont conduit à l'établissement d'un autre algorithme dit NLMS (*Normalized LMS*).

### 2.2.2 Algorithme NLMS

Afin de surmonter les deux problèmes cités précédemment, une normalisation du pas d'adaptation est réalisée. En effet la trace  $tr(R_x)$  de la matrice d'auto-corrélation du signal d'entrée est donnée par [4]:

$$tr(R_x) = L E\{|x(n)|^2\} \quad (2.21)$$

La quantité  $E\{|x(n)|^2\}$  est la puissance du signal d'entrée qui peut être estimée par :

$$\hat{E}\{|x(n)|^2\} = \frac{1}{L} \sum_{k=0}^{L-1} |x(n-k)|^2 = \frac{1}{L} \|\mathbf{x}(n)\|^2 \quad (2.22)$$

la condition de convergence en moyenne quadratique du LMS devient alors :

$$0 < \mu < \frac{2}{\|\mathbf{x}(n)\|^2} \quad (2.23)$$

Une méthode efficace d'incorporer cette condition dans l'équation d'adaptation de l'algorithme LMS est de faire varier le pas d'adaptation de sorte que :

$$\mu(n) = \frac{\beta}{\|\mathbf{x}(n)\|^2} \quad (2.24)$$

où  $\beta$  est le pas d'adaptation normalisé. En tenant compte de la condition de convergence, la valeur de  $\beta$  doit être comprise dans l'intervalle  $[0,2]$ .

En remplaçant dans l'équation (2.20) on aboutit à l'algorithme NLMS où l'équation d'adaptation s'exprime par :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \beta \frac{\mathbf{x}^*(n)}{\|\mathbf{x}(n)\|^2} e(n) \quad (2.25)$$

On note qu'avec cette procédure de normalisation, le problème d'amplification du bruit du gradient est également éliminé, vu que la séquence d'entrée est aussi normalisée par rapport à la puissance du signal d'entrée. Par contre un problème similaire se produirait dans le cas d'une faible puissance du signal d'entrée. Une solution pratique consiste en l'équation d'adaptation suivante :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \beta \frac{\mathbf{x}^*(n)}{\varepsilon + \|\mathbf{x}(n)\|^2} e(n) \quad (2.26)$$

où  $\varepsilon$  est une constante positive de faible valeur.

Malgré leur simplicité d'implémentation, la complexité arithmétique des algorithmes LMS et NLMS reste élevée. A cet effet certaines simplifications sont nécessaires dans des applications où le temps de calcul doit être minime. L'une de ces simplifications est la procédure de traitement par bloc proposée par *Clarck* et al. [16] qui a conduit à l'élaboration de l'algorithme dit BLMS (*Block LMS*).

### 2.2.3 Algorithme BLMS

Dans les algorithmes LMS et NLMS, la mise à jour des coefficients du filtre est effectuée à chaque nouvel échantillon. par contre dans cet algorithme les  $L$  coefficients du filtre sont maintenus constants pendant la durée de traitement de tout un bloc de  $N$  échantillons.

Au cours du  $s^{\text{ième}}$  bloc, chaque échantillon de sortie du filtre est donnée par :

$$y(sN+m) = \sum_{k=0}^{L-1} h_s(k) x(sN+m-k) = \mathbf{x}_{sN+m}^T \mathbf{h}_s \quad \text{pour } m = 1, \dots, N \quad (2.27)$$

où  $\mathbf{h}_s$  est le vecteur des coefficients du filtre à l'itération  $s$  :

$$\mathbf{h}_s = [h_s(0), \dots, h_s(L-1)]^T \quad (2.28)$$

et  $\mathbf{x}_{sN+m}$  le vecteur des échantillons d'entrée nécessaires pour l'évaluation de la sortie du filtre à l'instant  $sN+m$  :

$$\mathbf{x}_{sN+m} = [x(sN+m), \dots, x(sN+m-L+1)]^T \quad (2.29)$$

En notation vectorielle, on peut écrire :

$$\mathbf{y}_s = \boldsymbol{\chi}_s \mathbf{h}_s \quad (2.30)$$

où  $\mathbf{y}_s$  est le vecteur des sorties du filtre correspondant au bloc des  $N$  entrées à l'itération  $s$  :

$$\mathbf{y}_s = [y(sN), \dots, y(sN+N-1)]^T \quad (2.31)$$

Et  $\boldsymbol{\chi}_s$  une matrice de taille  $N \times L$  dont les lignes sont constituées par les transposées des vecteurs  $\mathbf{x}_i$  (pour  $i = sN, \dots, sN+N-1$ ) :

$$\boldsymbol{\chi}_s = \begin{bmatrix} x(sN) & x(sN-1) & \dots & x(sN-L+1) \\ x(sN+1) & x(sN) & \dots & x(sN-L+2) \\ \vdots & \vdots & \ddots & \vdots \\ x(sN+N-1) & x(sN+N-2) & \dots & x(sN+N-L) \end{bmatrix} \quad (2.32)$$

Le vecteur d'erreur sera exprimé par la différence entre le vecteur des sorties estimées et le bloc des sorties désirées :

$$\mathbf{e}_s = \mathbf{d}_s - \mathbf{y}_s \quad (2.33)$$

De façon similaire au LMS et par superposition sur tout le bloc, le gradient s'exprime par [17] :

$$\nabla_{\mathbf{h}}(s) = \sum_{k=0}^{N-1} \mathbf{x}_{sN+k}^* e(sN+k) = \boldsymbol{\chi}_s^H \mathbf{e}_s \quad (2.34)$$

Par conséquent, l'adaptation des coefficients du filtre s'effectue après le traitement de chaque bloc, suivant :

$$\mathbf{h}_{s+1} = \mathbf{h}_s + \mu \mathbf{x}_s^H \mathbf{e}_s \quad (2.35)$$

Le choix du pas d'adaptation  $\mu$  doit satisfaire les mêmes conditions de convergence que pour le LMS [15].

La complexité arithmétique du BLMS peut être réduite considérablement en utilisant une procédure de calcul de convolution basée sur la transformée de Fourier rapide (FFT) [18]. Cette nouvelle approche constitue un début pour l'implémentation des algorithmes de filtrage adaptatif dans le domaine fréquentiel.

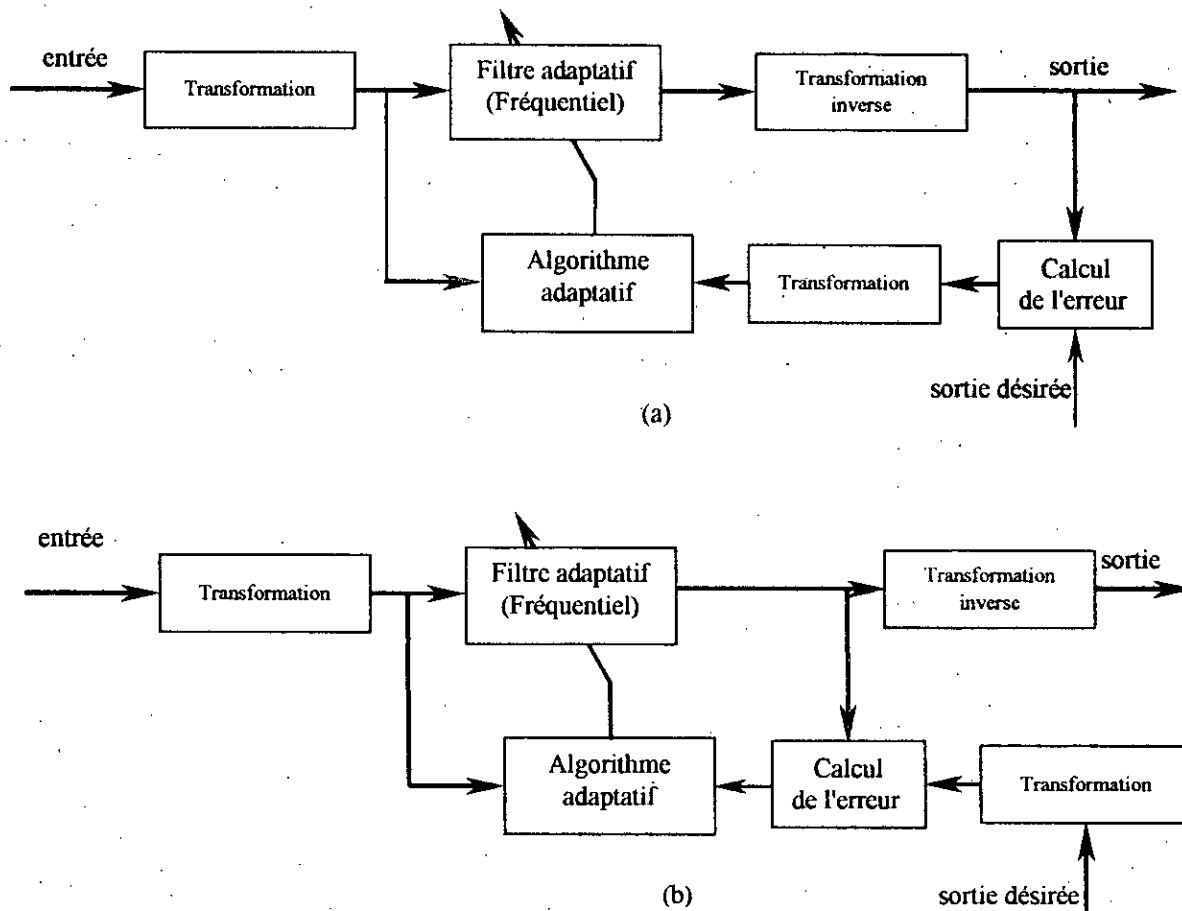
### 2.3 Algorithmes de filtrage adaptatif dans le domaine fréquentiel

L'opération de base dans les algorithmes de filtrage adaptatif dans le domaine fréquentiel est la transformation du signal d'entrée, par l'utilisation de la transformée de Fourier discrète, avant l'application du processus adaptatif. Comme montré dans la figure 2.3, cette transformation n'est pas adaptative et peut être vue comme étant une étape préliminaire indépendante des signaux d'entrée. On constate que deux configurations sont possibles : dans la première (Figure 2.3.a), le calcul de l'erreur est effectué dans le domaine temporel ensuite transformée vers le domaine fréquentiel. Par contre dans la deuxième forme (Figure 2.3.b), le calcul de l'erreur s'effectue dans le domaine fréquentiel, après transformation de la sortie désirée. Pour des algorithmes adaptatifs où la fonction d'erreur est linéaire ces deux configurations donnent des résultats identiques. Par contre, dans le cas où la fonction d'erreur n'est pas linéaire les deux formes précédentes aboutissent à des résultats différents [17].

L'implémentation dans le domaine fréquentiel des algorithmes de filtrage adaptatif offre essentiellement deux avantages [17] :

1. La réduction de la charge de calcul par le remplacement de la convolution et la corrélation temporelles par des convolutions ou corrélations rapides, basées sur la transformée de Fourier rapide (voir annexe B).
2. La décorrélation du signal d'entrée par l'application de la transformée de Fourier [19], permet d'utiliser un pas d'adaptation variable pour chaque coefficient du filtre. En effet, la vitesse de convergence des algorithmes dérivés de l'algorithme du gradient est inversement proportionnelle à la dispersion des valeurs propres de la matrice d'auto-corrélation du signal

d'entrée [3]. De plus ces valeurs propres correspondent approximativement à la densité spectrale de puissance du signal d'entrée à des points fréquentiels équidistants autour du cercle unité [20], [21]. Il est donc possible de compenser cette dispersion en utilisant, pour chaque coefficient du filtre, un pas d'adaptation inversement proportionnel à la densité spectrale de puissance du signal d'entrée. Par conséquent, une convergence quasi-uniforme pour tous les coefficients du filtre est obtenue, même en présence de larges variations de la puissance du signal d'entrée [1].



**Figure 2.3 :** Configuration générale d'un algorithme de filtrage adaptatif dans le domaine fréquentiel

### 2.3.1 Algorithme FBLMS

L'implémentation du BLMS en fréquence a été initialement proposée par *Ferrara* [22]. L'opération de filtrage est réalisée par une procédure de convolution rapide dans le domaine fréquentiel, (voir [20] et annexe B). Deux méthodes peuvent être utilisées : la procédure OLS (*Overlap-save*, i.e. recouvrement-sauvegarde) et OLA (*Overlap-add*, i.e. recouvrement-addition). Néanmoins la procédure OLS est la plus utilisée vu qu'elle nécessite moins de calculs comparée à la procédure OLA.

• **Equations de filtrage**

En utilisant la procédure OLS pour le calcul de la convolution, le bloc des  $N$  échantillons d'entrée devra être complété par les  $L-1$  échantillons précédents et les  $L$  coefficients du filtre par  $N-1$  zéros. De cette façon la matrice  $X_s$  définie dans (2.30) sera circulante. En utilisant la propriété de diagonalisation des matrices circulantes dans le domaine fréquentiel, on peut écrire [23]:

$$Y_s = X_s \odot H_s \tag{2.36}$$

$$\text{où } X_s = W_M [x(sN-L+1), \dots, x(sN-1), x(sN), \dots, x(sN+N-1)]^T \tag{2.37}$$

$$H_s = W_M \begin{bmatrix} h_s \\ \mathbf{O}_{(M-N) \times 1} \end{bmatrix} \tag{2.38}$$

Le symbole  $\odot$  représente le produit de Schur des vecteurs complexes (élément par élément).

$M$  est la plus petite puissance de 2 supérieure ou égale à  $L+N-1$ .

Les  $L-1$  premiers termes de la TFD inverse de  $Y_s$  correspondent à une convolution circulaire non désirée, alors que seuls les  $N$  derniers termes seront considérés vu que ceux-ci correspondent à la convolution linéaire entre le filtre et le bloc des entrées, (voir [22] et annexe B). la troncature de ces  $N$  derniers termes s'effectue par l'application d'une fenêtre rectangulaire  $f$  :

$$f = \begin{bmatrix} \mathbf{O}_{(L-1) \times (L-1)} & \mathbf{O}_{(L-1) \times N} \\ \mathbf{O}_{N \times (L-1)} & \mathbf{I}_{N \times N} \end{bmatrix} \tag{2.39}$$

alors la sortie du filtre est donnée par :

$$y_s = f W_M^{-1} (X_s \odot H_s) \tag{2.40}$$

L'erreur d'estimation peut être calculée, soit dans le domaine temporel par la relation (2.33) ou dans le domaine fréquentiel [17] :

$$e_s = d_s - f W_M^{-1} X_s \odot H_s \tag{2.41}$$

$$E_s = F (D_s - X_s \odot H_s) \tag{2.42}$$

où  $F$  est la matrice circulante associée à  $f$  i.e. ( $F = W_M f W_M^{-1}$ ) :

$$F = W_M \begin{bmatrix} \mathbf{O}_{(L-1) \times (L-1)} & \mathbf{O}_{(L-1) \times N} \\ \mathbf{O}_{N \times (L-1)} & \mathbf{I}_{N \times N} \end{bmatrix} W_M^{-1} \tag{2.43}$$

et  $D_s$  la transformée de Fourier discrète du bloc des réponses désirées:

$$D_s = W_M [O_{1 \times (L-1)}, d(sN), \dots, d(sN+N-1)]^T \quad (2.44)$$

### • Equations d'adaptation

L'équation d'adaptation est implémentée en fréquence en remarquant que le  $m^{\text{ième}}$  élément du gradient  $\nabla_s = \chi_s^H e_s$  s'exprime par :

$$\nabla_s(m) = \sum_{k=0}^{N-1} e(sN+k) x^*(sN+k-m); \quad \text{pour } m = 1, \dots, N-1 \quad (2.45)$$

Les éléments du gradient sont donnés par une inter-corrélation entre l'erreur d'estimation et l'entrée du filtre. En procédant de la même manière que pour la convolution, (i.e. précéder le bloc d'entrée par les  $L-1$  échantillons précédents de sorte que la matrice des données devienne circulante et précéder également le vecteur d'erreur par  $N-1$  zéros), cette inter-corrélation peut être efficacement calculée dans le domaine fréquentiel [17], [23], en :

(1) Calculant la TFD de l'erreur  $e_s$  précédée de  $L-1$  zéros ;

$$E_s = W_M \begin{bmatrix} O_{(L-1) \times 1} \\ e_s \end{bmatrix} \quad (2.46)$$

(2) Effectuant le produit de Schur :

$$\tilde{\nabla}_s = X_s^* \odot E_s \quad (2.47)$$

(3) Remarquant que les  $L$  premiers termes de la TFD inverse de  $\tilde{\nabla}_s$  correspondent à l'inter-corrélation linéaire  $\nabla_s$ , tandis que les  $N-1$  derniers termes correspondent à une inter-corrélation circulaire et ne doivent donc pas être pris en compte [17]. La troncature des éléments correspondant à l'inter-corrélation linéaire s'effectue par l'application d'une fenêtre rectangulaire  $g$  :

$$g = \begin{bmatrix} I_{L \times L} & O_{L \times (N-1)} \\ O_{(N-1) \times L} & O_{(N-1) \times (N-1)} \end{bmatrix} \quad (2.48)$$

L'équation d'adaptation dans le domaine fréquentiel s'écrit alors comme suit :

$$H_{s+1} = H_s + \mu G (X_s^* \odot E_s) \quad (2.49)$$

où  $G$  est la matrice circulante associée à  $g$  i.e. ( $G = W_M g W_M^{-1}$ ) :

$$G = W_M \begin{bmatrix} I_{L \times L} & O_{L \times (N-1)} \\ O_{(N-1) \times L} & O_{(N-1) \times (N-1)} \end{bmatrix} W_M^{-1} \quad (2.50)$$

Comme montré par *Mansour* et *Gray* [26], la charge de calcul peut être réduite sans affecter la convergence de l'algorithme en négligeant l'étape (3) du calcul du gradient. Ceci conduit à l'algorithme FBLMS dit *non constraint* dans lequel les termes de repliement du produit  $(X_s^* \odot E_s)$  sont maintenus. L'équation d'adaptation s'exprime alors par :

$$H_{s+1} = H_s + \mu X_s^* \odot E_s \quad (2.51)$$

Par opposition, l'algorithme défini par l'équation (2.49) est appelé algorithme FBLMS *constraint*, dans lequel les termes de repliement du produit  $(X_s^* \odot E_s)$  sont éliminés par le projecteur  $G$ .

- **Normalisation dans le domaine fréquentiel**

Les coefficients du filtre mis à jour dans le domaine fréquentiel par les équations (2.49 et 2.51) sont adaptés indépendamment les uns des autres. Il est donc possible d'utiliser pour chaque coefficient un pas d'adaptation approprié. Ceci se traduit par la prémultiplication (produit de Schur) du terme correctif  $(X_s^* \odot E_s)$  par un vecteur  $T_s$ , de dimension  $M$ , mis à jour à chaque itération. l'équation de mise à jour du filtre devient alors, pour l'algorithme *constraint* :

$$H_{s+1} = H_s + \mu G(T_s \odot X_s^* \odot E_s) \quad (2.52)$$

et pour le *non constraint* :

$$H_{s+1} = H_s + \mu T_s \odot X_s^* \odot E_s \quad (2.53)$$

Cette nouvelle opération appelée *normalisation* va permettre d'améliorer la rapidité de convergence de l'algorithme. En effet, le choix d'un pas d'adaptation inversement proportionnel à la puissance du signal d'entrée à la fréquence associée au coefficient du filtre permet de compenser la dispersion énergétique. En conséquence, une convergence quasi-uniforme pour tous les coefficients du filtre adaptatif est obtenue [1].

En pratique, la puissance du signal d'entrée est une inconnue qui varie dans le temps. Une estimation régulière de cette quantité est alors nécessaire. L'une des estimations possibles s'exprime par la récurrence suivante [27]:

$$P_s(k) = \gamma P_{s-1}(k) + (1-\gamma) |X_s(k)|^2 \quad (2.54)$$

où  $X_s(k)$  est la  $k^{\text{ième}}$  composante de la TFD du vecteur d'entrée, et le facteur d'oubli  $\gamma$  est un coefficient permettant un lissage exponentiel de l'énergie et dont la valeur se situe dans l'intervalle  $[0,1]$ .



Les éléments du vecteur de normalisation  $T_s$  sont alors donnés par :

$$T_s = [P_s^{-1}(1), \dots, P_s^{-1}(M)] \quad (2.55)$$

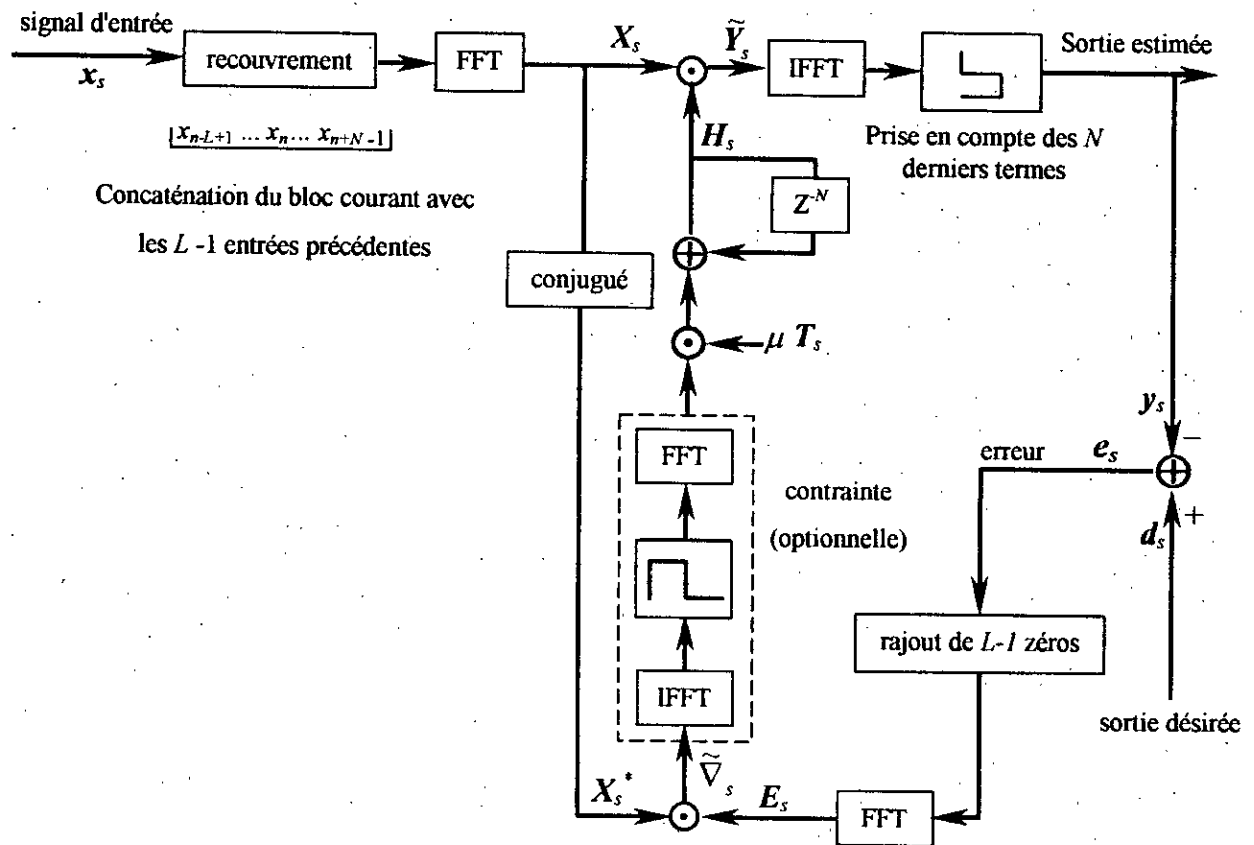


Figure 2.4 : Algorithme FBLMS

### 2.3.2 Algorithme MDF

Dans la version originale de l'algorithme FBLMS présenté ci-dessus, les transformées sont de longueur  $M \geq L+N-1$ . Ceci peut poser des problèmes d'implémentation en temps réel, notamment pour les très longues réponses impulsionnelles. En effet, le calcul de très longues FFTs nécessite un espace mémoire considérable et entraîne de grandes erreurs de d'arrondi. D'autre part, le calcul de convolution avec de très longues séquences entraînerait des retards prohibitifs, parfois inacceptables.

L'idée de base de cet algorithme, introduit par *Soo* et *Pang* [28], est la segmentation de la réponse impulsionnelle en  $K$  segments de longueur  $P$  de telle sorte que  $L=KP$ , et d'utiliser de courtes FFTs de taille  $M$ , où  $M$  est la plus petite puissance de 2 supérieure ou égale à  $N+P-1$ . La convolution dans le domaine temporel se traduit alors par la somme de  $K$  convolutions partielles entre les  $K$  segments de la réponse impulsionnelle et la séquence du signal d'entrée qui est

également segmentée en  $K$  sous blocs. Ceci conduit à l'algorithme dit à *délais multiples* ; MDF (*Multidelay Frequency-Domain Adaptive Filter*).

- **Equations de convolution**

Considérons la réponse impulsionnelle de longueur  $L=KP$  :

$$\mathbf{h}_s = [h_s(0), \dots, h_s(P-1), \dots, h_s(2P-1) \dots, \dots, h_s(KP-1)]^T \quad (2.56)$$

Après segmentation en  $K$  sous blocs de longueur  $P$  :

$$\begin{aligned} \mathbf{h}_s &= [[h_s(0), \dots, h_s(P-1)] [h_s(P), \dots, h_s(2P-1)] \dots [h_s(KP-P), \dots, h_s(KP-1)]]^T \\ &= [\mathbf{h}_s^{(0)}, \mathbf{h}_s^{(1)} \dots, \mathbf{h}_s^{(K-1)}]^T \end{aligned} \quad (2.57)$$

On définit  $\mathbf{h}_s^{(k)}$  comme étant le  $k^{\text{ième}}$  segment de la réponse impulsionnelle à l'instant  $sN$ . Les éléments de  $\mathbf{h}_s^{(k)}$  sont donnés par :

$$\mathbf{h}_s^{(k)} = [h_s(kP), \dots, h_s(kP+P-1)]^T \quad \text{pour } k = 0, \dots, K-1. \quad (2.58)$$

En effectuant la même opération pour la séquence des entrées dont la longueur est identique à celle de la réponse impulsionnelle, i.e.  $L=KP$  :

$$\begin{aligned} \mathbf{x}_{sN} &= [x(sN), \dots, x(sN-P+1), \dots, x(sN-2P+1) \dots, \dots, x(sN-KP+1)]^T \\ &= [[x(sN), \dots, x(sN-P+1)] [x(sN-P), \dots, x(sN-2P+1)] \dots [x(sN-KP+P), \dots, x(sN-KP+1)]]^T \\ &= [\mathbf{x}_{sN}^{(0)}, \mathbf{x}_{sN}^{(1)} \dots, \mathbf{x}_{sN}^{(K-1)}]^T \end{aligned} \quad (2.59)$$

Le vecteur  $\mathbf{x}_{sN}^{(k)}$  est le  $k^{\text{ième}}$  sous bloc du vecteur temporel des échantillons d'entrée  $\mathbf{x}_{sN}$ . C'est st un vecteur de longueur  $P$  donné par :

$$\mathbf{x}_{sN}^{(k)} = [x(sN-kP), \dots, x(sN-kP+P-1)]^T, \quad \text{pour } k = 0, \dots, K-1. \quad (2.60)$$

La convolution à l'instant  $sN$  s'écrit alors :

$$\begin{aligned} y(sN) &= h_s(0) x(sN) + h_s(1) x(sN-1) + \dots + h_s(P-1) x(sN-P+1) + \\ &\quad h_s(P) x(sN-P) + h_s(P+1) x(sN-P-1) + \dots + h_s(2P-1) x(sN-2P+1) + \\ &\quad \dots + \\ &\quad h_s(KP-P) x(sN-KP+P) + h_s(KP-P+1) x(sN-KP+P-1) + \dots + h_s(KP-1) x(sN-KP+1) \end{aligned}$$

En utilisant la forme vectorielle :

$$\begin{aligned} y(sN) &= \mathbf{h}_s^{(0)T} \mathbf{x}_{sN}^{(0)} + \mathbf{h}_s^{(1)T} \mathbf{x}_{sN}^{(1)} + \dots + \mathbf{h}_s^{(K-1)T} \mathbf{x}_{sN}^{(K-1)} \\ &= \mathbf{h}_s^{(k)T} \mathbf{x}_{sN}^{(k)} \end{aligned} \quad (2.61)$$

On constate que pour chaque nouvelle entrée, la convolution de longueur  $L$  est décomposée en  $K$  convolutions partielles de longueur  $P$ .

En introduisant la procédure de traitement des données par blocs de  $N$  échantillons et de façon similaire à (2.30), on peut écrire :

$$y_s = \sum_{k=1}^{K-1} \mathbf{h}_s^{(k)T} \boldsymbol{\chi}_s^{(k)} \quad (2.62)$$

où  $\boldsymbol{\chi}_s^{(k)}$  est la matrice des données correspondant à la  $k^{\text{ème}}$  convolution partielle :

$$\boldsymbol{\chi}_s^{(k)} = [\mathbf{x}_{sN}^{(k)}, \mathbf{x}_{sN+1}^{(k)}, \dots, \mathbf{x}_{sN+N-1}^{(k)}]^T, \text{ pour } k=1, \dots, K. \quad (2.63)$$

En vertu du principe de superposition et par l'utilisation de la procédure OLA pour le calcul de chaque convolution partielle dans le domaine fréquentiel, la sortie du filtre s'écrit :

$$\mathbf{y}_s = \mathbf{f} \mathbf{W}_M^{-1} \left( \sum_{k=0}^{K-1} \mathbf{X}_s^{(k)} \odot \mathbf{H}_s^{(k)} \right) \quad (2.64)$$

où  $\mathbf{H}_s^{(k)}$  le vecteur de longueur  $M$ , transformé de Fourier de  $\mathbf{h}_s^{(k)}$  complété par  $(N-1)$  zéros :

$$\mathbf{H}_s^{(k)} = \mathbf{W}_M \begin{bmatrix} \mathbf{h}_s^{(k)} \\ \mathbf{0}_{(N-1) \times 1} \end{bmatrix} \quad (2.65)$$

et  $\mathbf{X}_s^{(k)}$  le vecteur de longueur  $M$  transformé de Fourier de  $\mathbf{x}_{sN}^{(k)}$  complété par  $(P-1)$  échantillons précédents :

$$\mathbf{X}_s^{(k)} = \mathbf{W}_M [x(sN-kP-P+1), \dots, x(sN-kP-1), x(sN-kP), \dots, x(sN-kP+N-1)]^T \quad (2.66)$$

Le symbole  $\odot$  représente le produit de Schur des vecteurs complexes (élément par élément). et  $\mathbf{f}$  est une fenêtre rectangulaire qui permet d'isoler les  $N$  derniers termes qui correspondent à la convolution linéaire recherchée.  $\mathbf{f}$  est définie de la même manière que dans (2.39).

A partir de ce résultat le vecteur d'erreur s'exprime par :

$$\mathbf{e}_s = \mathbf{d}_s - \mathbf{f} \mathbf{W}_M^{-1} \left( \sum_{k=0}^{K-1} \mathbf{X}_s^{(k)} \odot \mathbf{H}_s^{(k)} \right) \quad (2.67)$$

L'organigramme décrivant cette méthode de calcul de la sortie du filtre est montré dans la figure 2.5.

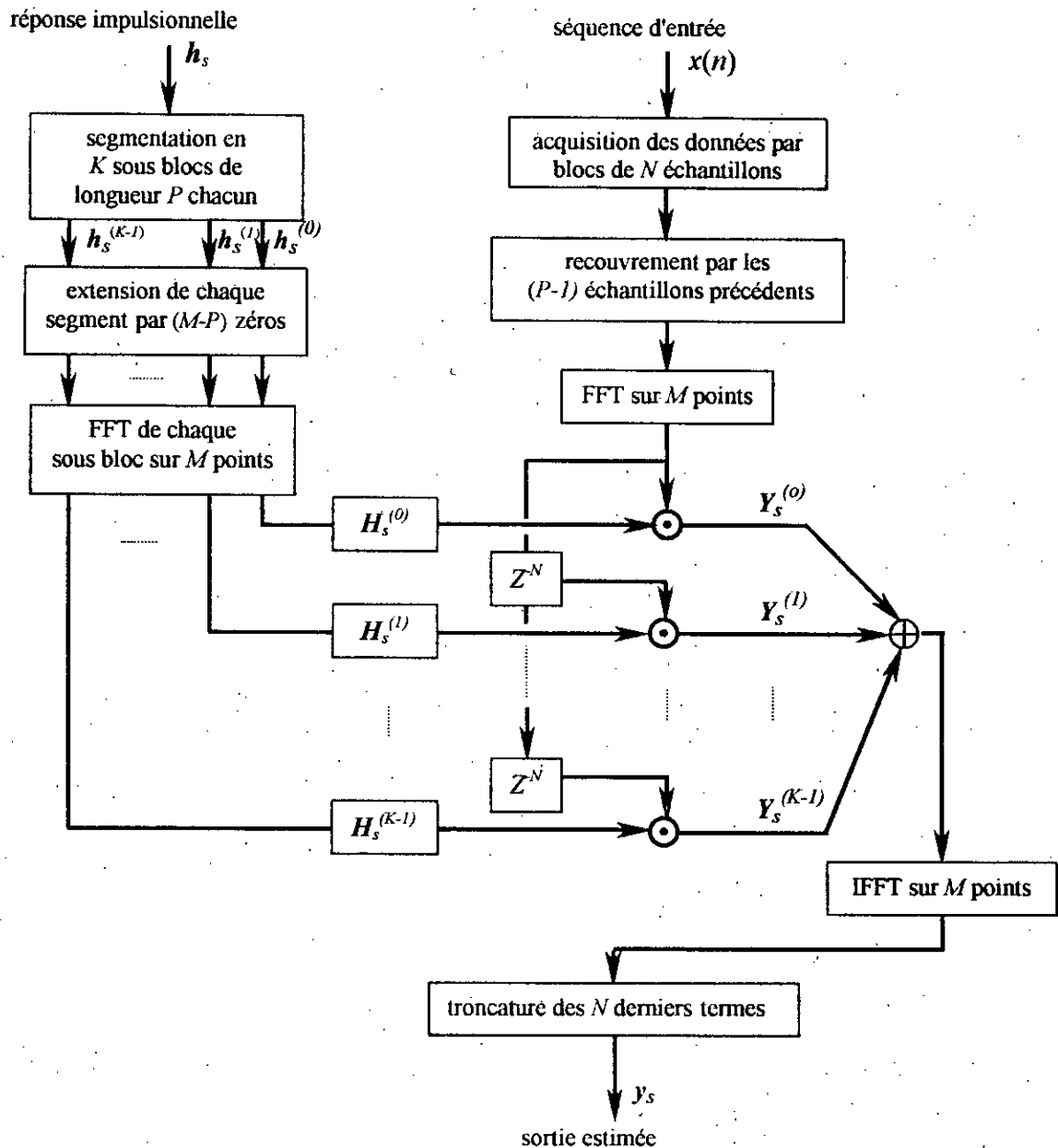


Figure 2.5 : Convolution rapide et segmentation

• Equations d'adaptation

En vertu du principe de superposition et de façon similaire à l'algorithme FBLMS, le calcul du gradient correspondant au  $k^{\text{ème}}$  segment de la réponse impulsionnelle s'effectue par le calcul de l'inter-corrélation entre le  $k^{\text{ème}}$  sous-bloc de la séquence d'entrée et le vecteur d'erreur. Par conséquent les équations d'adaptation dans le domaine fréquentiel des coefficients de chaque segment du filtre s'expriment par :

pour le cas contraint :

$$H_{s+1}^{(k)} = H_s^{(k)} + \mu G (T_s \odot (X_s^{(k)})^* \odot E_s) \quad (2.68)$$

et pour le cas non contraint :

$$H_{s+1}^{(k)} = H_s^{(k)} + \mu (T_s \odot (X_s^{(k)})^* \odot E_s) \quad (2.69)$$

où

$$G = W_M \begin{bmatrix} I_{P \times P} & O_{P \times (N-1)} \\ O_{(N-1) \times P} & O_{(N-1) \times (N-1)} \end{bmatrix} W_M^{-1} \quad (2.70)$$

$T_s$  est le vecteur de normalisation dans le domaine fréquentiel dont les éléments sont calculés de la même manière que dans le FBLMS.

**Remarque :**

Afin de minimiser la charge de calcul, un choix judicieux est de prendre  $N = P$ . Dans ce cas  $X_s^{(k)} = X_{s-k}^{(0)}$ ,  $0 < k < K$ . Par conséquent, à chaque itération, on évalue seulement la FFT du bloc du signal d'entrée le plus récent, i.e.  $X_s^{(0)}$ .

### 2.3.3 Algorithme GMDF $\alpha$

Dans ce qui précède, il est implicitement supposé que la taille du bloc d'entrée et le décalage entre deux sections successives  $x_s^{(k)}$  sont identiques, i.e. les sections du signal d'entrée sont décalées de  $N$  échantillons à chaque itération. Dans l'algorithme GMDF $\alpha$  (*Generalized Multidelay Frequency-Domain Adaptive Filter  $\alpha$* ) proposé par *Ait Amrane et al.* [30], en plus de la segmentation de la réponse impulsionnelle, une souplesse supplémentaire est introduite en permettant un recouvrement entre les blocs d'entrée successifs et cela en choisissant un décalage de  $R \leq N$  échantillons. Par souci de simplicité, nous supposons par la suite que  $R$  est une fraction entière de  $N$ , i.e.  $R = [N/\alpha]$  (le symbole  $[x]$  signifie la partie entière de  $x$ ), et  $\alpha$  est un entier appelé *facteur de recouvrement* (appelé également *facteur de suréchantillonnage*). Le choix de  $R < N$  est de peu d'intérêt pour une convolution rapide (annexe B), par contre l'utilisation d'un grand facteur de recouvrement apparaît essentiel dans le contexte du filtrage adaptatif car, il permet une adaptation plus fréquente des coefficients du filtre (chaque  $R$  échantillons au lieu de  $N$ ). En conséquence, de meilleures performances tant du point de vue vitesse de convergence que capacités de poursuite peuvent être atteintes.

### • Equations de filtrage

A chaque itération,  $R$  nouveaux échantillons du signal d'entrée et du signal désiré seront traités :

$$\mathbf{x}_{sR} = [x(sR), x(sR+1), \dots, x(sR+R-1)]^T \quad (2.71)$$

$$\mathbf{d}_{sR} = [d(sR), x(sR+1), \dots, d(sR+R-1)]^T \quad (2.72)$$

Afin de former des blocs de  $N$  échantillons, ces nouveaux échantillons sont complétés par les  $N-R$  échantillons précédents :

$$\mathbf{x}_s = [x(sR-(\alpha-1)R), \dots, x(sR-1), x(sR), \dots, x(sR+R-1)]^T \quad (2.73)$$

$$\mathbf{d}_s = [d(sR-(\alpha-1)R), \dots, d(sR-1), d(sR), \dots, d(sR+R-1)]^T \quad (2.74)$$

Les équations de filtrage dans l'algorithme GMDF $\alpha$  s'obtiennent et s'expriment de la même manière que dans l'algorithme MDF développé précédemment. La convolution du bloc d'entrée par la réponse impulsionnelle est donnée alors par :

$$\mathbf{y}_s = \mathbf{f} \mathbf{W}_M^{-1} \left( \sum_{k=0}^{K-1} \mathbf{X}_s^{(k)} \odot \mathbf{H}_s^{(k)} \right) \quad (2.75)$$

où  $\mathbf{X}_s^{(k)}$  est le vecteur transformé de Fourier du  $k^{\text{ème}}$  segment du bloc d'entrée complété par les  $(P-1)$  échantillons précédents :

$$\mathbf{X}_s^{(k)} = \mathbf{W}_M [x(sR-(k+1)P-(\alpha-1)R+1), \dots, x(sR-kP-1), x(sR-kP), \dots, x(sR-kP+R-1)]^T \quad (2.76)$$

$M$  étant la plus petite puissance de 2 supérieure ou égale à  $P+N-1$ .

$\mathbf{H}_s^{(k)}$  le vecteur de longueur  $M$  transformé de Fourier de  $\mathbf{h}_s^{(k)}$  complété par  $(N-1)$  zéros :

$$\mathbf{H}_s^{(k)} = \mathbf{W}_M \begin{bmatrix} \mathbf{h}_s^{(k)} \\ \mathbf{O}_{(N-1) \times 1} \end{bmatrix} \quad (2.77)$$

la fenêtre  $\mathbf{f}$  est définie de la même manière que dans (2.39) et permet d'isoler les termes correspondant à la convolution linéaire désirée.

### • Equations d'adaptation

Les équations d'adaptation des coefficients du filtre s'obtiennent et s'expriment de la même manière que dans le MDF. Les versions contrainte et non contrainte de l'algorithme GMDF $\alpha$  sont alors données respectivement par :

$$\left\{ \begin{array}{l} H_{s+1}^{(k)} = H_s^{(k)} + \mu G ( T_s \odot (X_s^{(k)})^* \odot E_s ) \\ H_{s+1}^{(k)} = H_s^{(k)} + \mu T_s \odot (X_s^{(k)})^* \odot E_s ) \end{array} \right. \quad (2.78)$$

$$\left\{ \begin{array}{l} H_{s+1}^{(k)} = H_s^{(k)} + \mu G ( T_s \odot (X_s^{(k)})^* \odot E_s ) \\ H_{s+1}^{(k)} = H_s^{(k)} + \mu T_s \odot (X_s^{(k)})^* \odot E_s ) \end{array} \right. \quad (2.79)$$

• **Reconstitution de la sortie par WOLA**

La différence essentielle entre le MDF et le GMDF $\alpha$  réside dans la façon de reconstruire le signal de sortie. Les blocs successifs d'entrée se recouvrent plus que nécessaire, il en résulte qu'à un instant donné  $n$  pour la sortie, le calcul aura été effectué  $\alpha$  fois. La méthode d'addition – recouvrement pondéré (*Weighted overlap add WOLA*) (voir annexe B et [30]) consiste à additionner avec la pondération appropriée, les échantillons de  $\alpha$  blocs de sortie successifs qui correspondent au même instant pour le signal de sortie final.

Soit  $g$  une fenêtre de reconstitution de longueur  $N+P-1$ , permettant à chaque itération d'isoler les  $N$  termes de la sortie OLS qui correspondent à une convolution linéaire, et satisfaisant la condition de normalisation :

$$\sum_{i=0}^{\alpha-1} g(n-iR) = 1, \text{ pour } n = P, \dots, N+P-1 \quad (2.80)$$

avec:  $g = [0_{1 \times (P-1)}, g(1), g(2), \dots, g(N)]^T$ .

On définit alors le bloc de sortie pondéré par :

$$\bar{y}_s = [0_{N \times (P-1)} \ I_{N \times N}] g \odot \tilde{y}_s \quad (2.81)$$

où  $\tilde{y}_s$  est obtenu suivant la relation (2.62).

Le bloc de sortie final s'obtient par addition-recouvrement des  $\alpha$  blocs pondérés successifs :

$$y(sR+n) = \sum_{j=0}^{\alpha-1} \bar{y}((s-j)R+P-1+n) \quad (2.82)$$

La structure globale de l'algorithme GMDF $\alpha$  est schématisée par la figure 2.6.

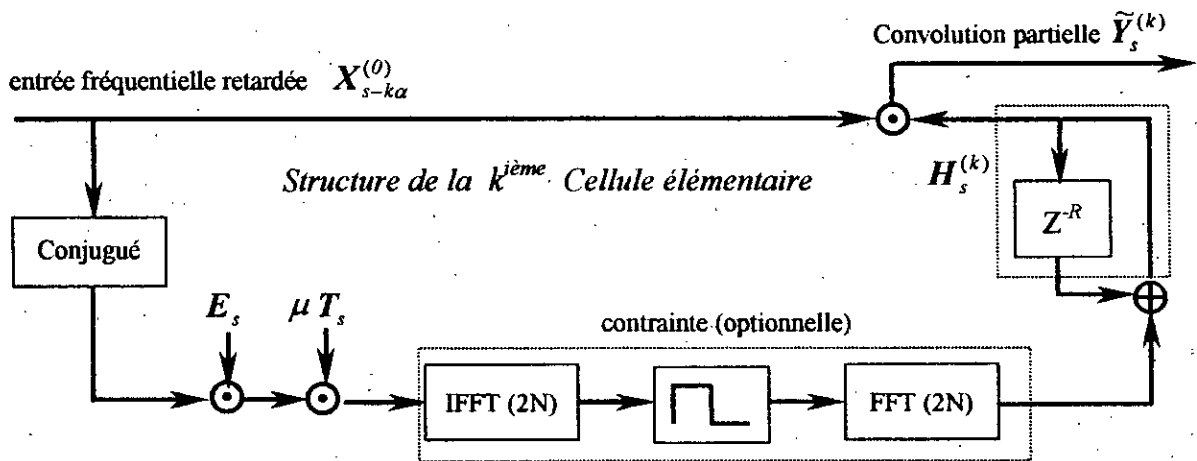
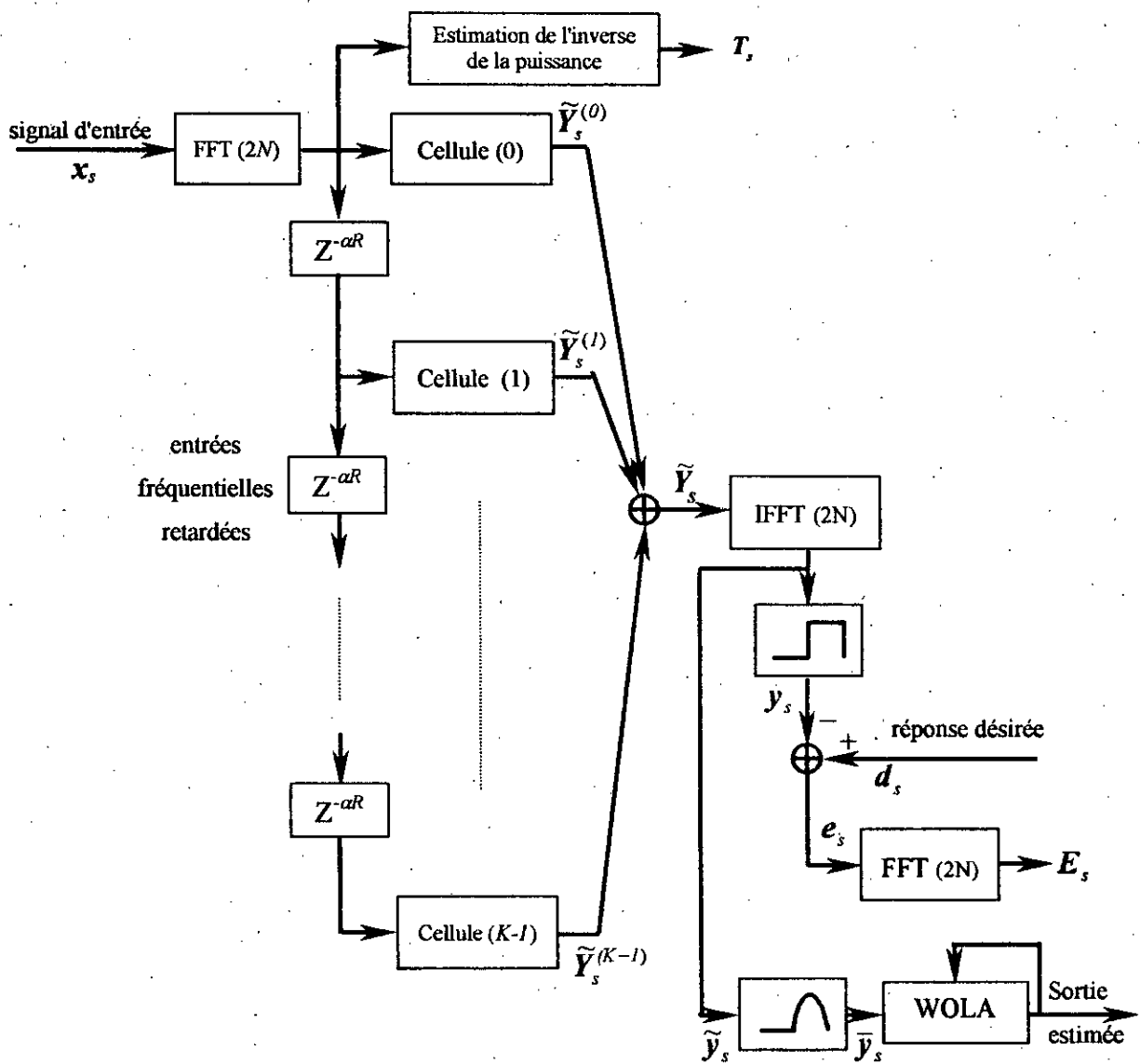


Figure 2.6 : Organigramme structurel de l'algorithme GMDF $\alpha$



## 2.4 Implémentation de l'algorithme GMDF $\alpha$

Nous allons maintenant présenter les différentes étapes permettant une implémentation à faible coût de calcul du filtre adaptatif basé sur l'algorithme GMDF $\alpha$ . La réponse impulsionnelle est de taille  $L=KP$  et le bloc des entrées est fixé à  $N=P$  échantillons. Les transformées de Fourier seront alors effectuées sur  $M \geq 2N-1$  points.

A chaque itération  $s$ , les différentes étapes de l'algorithme consistent en :

- **Acquisition du vecteur  $x_{sR}$  des  $R$  nouveaux échantillons :**

$$x_{sR} = [x(sR), \dots, x(sR+R-1)]^T \quad (2.83)$$

- **Recouvrement des  $R$  nouveaux échantillons d'entrée par les  $2N-R$  précédents.**

$$x_s = [x(sR-2N+R), \dots, x(sR-1), x(sR), \dots, x(sR+R-1)]^T \quad (2.84)$$

- **Evaluation du bloc fréquentiel d'entrée courant.**

$$X_s^{(0)} = W_{2N} [x(sR-2N+R), \dots, x(sR-1), x(sR), \dots, x(sR+R-1)]^T \quad (2.85)$$

- **Estimation de la puissance du signal d'entrée : pour  $i=1, \dots, 2N$**

$$P_s(i) = \gamma P_{s-1}(i) + (1-\gamma) |X_s^{(0)}(i)|^2 \quad (2.86)$$

- **Mise à jour des coefficients de normalisation.**

$$T_s = [P_s^{-1}(1), \dots, P_s^{-1}(2N)] \quad (2.87)$$

- **Rafraîchissement de  $K-1$  entrées fréquentielles retardées :**

$$X_s^{(k)} = X_{s-k\alpha}^{(0)} \quad (2.88)$$

- **Evaluation du bloc de sorties temporelles court terme :**

$$\bar{y}_s = [O_{N \times N} \quad O_{N \times N}] g \odot W_{2N}^{-1} \left( \sum_{k=0}^{K-1} X_s^{(k)} \odot H_s^{(k)} \right) \quad (2.89)$$

- **Reconstruction par WOLA des  $R$  échantillons de sortie. pour  $n=1, \dots, R$  :**

$$y(sR+n) = \sum_{j=0}^{\alpha-1} \bar{y}((s-j)R+n) \quad (2.90)$$

- **Evaluation du bloc fréquentiel de sortie désirée :**

$$D_s = W_{2N} [d(sR-2N+R), \dots, d(sN-1), d(sN), \dots, d(sN+R-1)]^T \quad (2.91)$$

- **Evaluation du signal d'erreur en fréquence.**

$$E_s = W_{2N} \begin{bmatrix} O_{N \times N} & O_{N \times N} \\ O_{N \times N} & I_{N \times N} \end{bmatrix} W_{2N}^{-1} \left( D_s - \sum_{k=0}^{K-1} X_s^{(k)} \odot H_s^{(k)} \right) \quad (2.92)$$

- **Evaluation du terme correctif du  $k^{\text{ième}}$  segment du filtre.**

$$\nabla_s^{(k)} = T_s \odot (X_s^{(k)})^* \odot E_s \quad (2.93)$$

- **Contrainte (optionnelle).**

$$\nabla_s^{(k)} = W_{2N} \begin{bmatrix} I_{N \times N} & O_{N \times N} \\ O_{N \times N} & O_{N \times N} \end{bmatrix} W_{2N}^{-1} (T_s \odot (X_s^{(k)})^* \odot E_s) \quad (2.94)$$

- **Mise à jour du  $k^{\text{ième}}$  segment du filtre.**

$$H_{s+1}^{(k)} = H_s^{(k)} + \mu \nabla_s^{(k)} \quad (2.95)$$

## 2.5 Etude de la complexité arithmétique de l'algorithme GMDF $\alpha$

La complexité arithmétique de l'algorithme GMDF $\alpha$  est fortement conditionnée par le nombre de FFTs. La version contrainte nécessite  $(2K+3)$  FFTs réelles de taille  $2N$  par contre la version non contrainte ne nécessite que 3 FFTs réelles de taille  $2N$ , ce qui constitue un gain considérable en charge de calcul. L'évaluation globale du nombre d'opérations réelles nécessaires par échantillon est donnée dans le tableau (2.1). Le nombre d'opérations nécessaires pour le calcul d'une FFT correspond à l'algorithme Split-radix [31]. On constate que le charge de calcul est proportionnelle au facteur de recouvrement.

	Multiplications réelles / échantillon	Additions réelles / échantillon
Non contraint	$\alpha [10K+3 \log_2(N) - 3]$	$\alpha [8K+9 \log_2(N)-10]$
Contraint	$\alpha [(2K+3) \log_2(N)+4K - 3]$	$\alpha [(6K+9) \log_2(N) - 2K - 10]$

**Tableau 2.1** [1] : *Complexité arithmétique de l'algorithme GMDF $\alpha$*

Cette complexité est comparée par rapport à celle du NLMS, en nombre de multiplications et d'additions réelles (figure 2.7), pour des réponses impulsionnelles de différentes longueurs. On remarque que le gain en charge de calcul croit de façon significative avec la taille de la réponse impulsionnelle. D'autre part, l'augmentation du nombre de segments de la réponse impulsionnelle engendre une augmentation considérable de la complexité arithmétique. En revanche, la segmentation reste toujours attractive pour de très longues réponses.

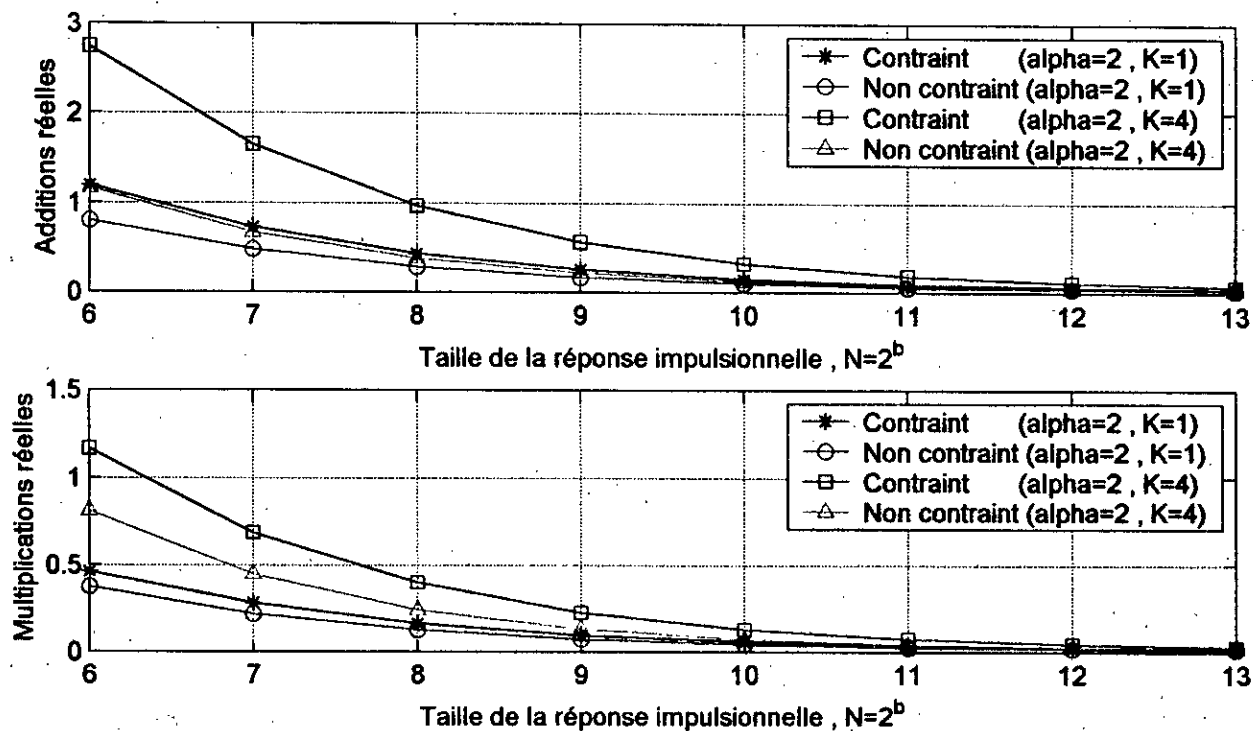


Figure 2.7 : Complexité de l'algorithme  $GMDF\alpha$  par rapport au NLMS

## 2.6 Conclusion

Dans ce chapitre, nous avons présenté les différents algorithmes de filtrage adaptatif susceptibles d'être appliqués à l'annulation d'écho acoustique. L'analyse de l'algorithme  $GMDF\alpha$  montre qu'il couvre toute une famille d'algorithmes fréquentiels et constitue une généralisation des algorithmes qui l'ont précédé :

- En l'absence de segmentation ( $K=1$ ), pour  $\alpha=1$ , cet algorithme correspond à l'algorithme FBLMS.
- En présence de segmentation ( $K \neq 1$ ), pour  $\alpha=1$ , cet algorithme correspond à l'algorithme à délais multiples (MDF).
- En absence de normalisation fréquentielle et de segmentation, pour  $\alpha=L$ , la version contrainte de cet algorithme correspond à une implémentation en fréquence du LMS avec moyennage du gradient sur  $L$  points

L'étude d'une caractéristique importante de ces algorithmes telle la complexité arithmétique a montré que le  $GMDF\alpha$  offre de meilleurs avantages pour une application où les réponses impulsionnelles sont très longues et où le temps de calcul doit être le plus faible possible.

# Chapitre 3

## Simulations et résultats

Dans ce chapitre nous allons présenter les résultats des simulations effectuées pour l'analyse des performances de l'algorithme GMDF $\alpha$ . Afin de mettre en évidence les propriétés adaptatives de l'algorithme, ces simulations ont été réalisées dans différents contextes. En premier lieu, les simulations ont été effectuées sur des signaux stationnaires permettant de discuter l'influence des différents paramètres sur la convergence de l'algorithme. En deuxième lieu, nous évaluerons les performances dans des situations où une non-stationnarité est introduite, soit par les signaux d'entrée (parole) ou par le système (canal variable).

Dans toutes les simulations les valeurs initiales de la réponse impulsionnelle et de la matrice de normalisation en fréquence sont nulles.

### 3.1 Modèle de simulation et notations

Afin d'analyser la convergence de l'algorithme nous adopterons le modèle de simulation représenté par la figure 3.1. Nous supposerons que la sortie désirée  $d(n)$  correspond à la sortie d'un système linéaire et stationnaire dont la réponse impulsionnelle  $h$  est excitée par le même signal d'entrée  $x(n)$  que le filtre adaptatif. Afin d'étudier l'influence du bruit additif sur la convergence de l'algorithme, on superpose à la sortie du filtre un bruit  $b(n)$ , gaussien, de variance  $\sigma^2$  et indépendant de  $x(n)$ .

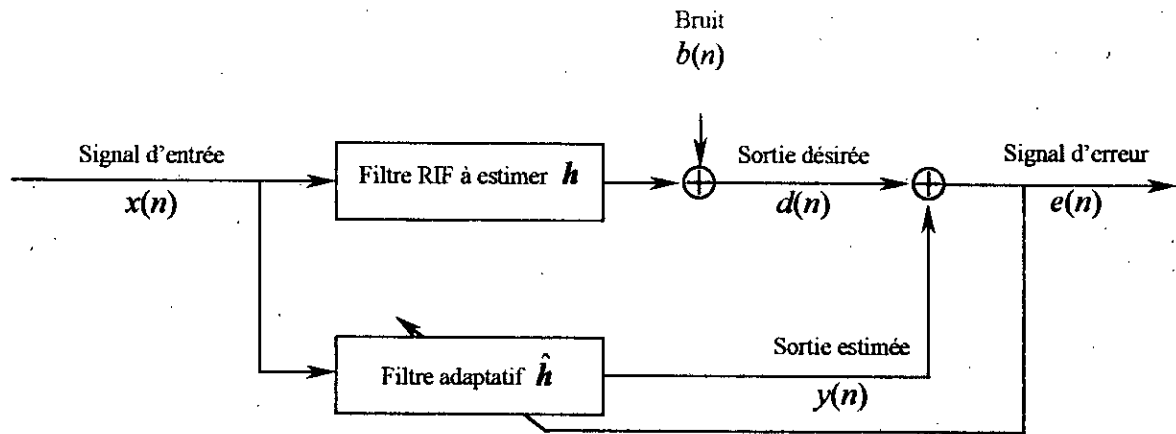


Figure 3.1 : Modèle de simulation et notations

### 3.2 Grandeurs décrivant les performances

Pour analyser les performances de l'algorithme, on utilisera les quantités suivantes comme critères d'évaluation :

#### 3.2.1 Erreur Quadratique Moyenne (EQM)

Cette quantité représente la puissance du signal "erreur" :

$$\text{EQM}(s) = E\{|e(s)|^2\} = \frac{1}{N} \sum_{k=1}^N e_s^2(k)$$

On appellera Erreur Quadratique Moyenne Résiduelle (EQMR) la valeur de l'EQM après convergence. Ce critère sera utilisé pour l'étude de la convergence de l'algorithme dans le cas des signaux stationnaires.

#### 3.2.2 Echo Return Loss Enhancement (ERLE)

Cette quantité représente le taux d'estimation du signal désiré :

$$\text{ERLE}(s) = \frac{E\{|d(s)|^2\}}{E\{|e(s)|^2\}} = \frac{\sum_{k=1}^N d_s^2(k)}{\sum_{k=1}^N e_s^2(k)}$$

On note que cette quantité converge vers le rapport signal sur bruit (RSB) du signal désiré. En effet, le RSB du signal désiré étant défini par :

$$\text{RSB} = \frac{E\{|d(n)|^2\}}{E\{|b(n)|^2\}}$$

La convergence de l'algorithme adaptatif vers la réponse impulsionnelle du système à identifier fait que l'erreur  $e(n)$  sera pratiquement égale au bruit  $b(n)$  d'où  $ERLE(\infty) \cong RSB$ .

Ce critère sera utilisé comme grandeur d'évaluation des algorithmes dans une situation d'annulation d'écho acoustique.

### 3.3 Analyse des performances dans un contexte stationnaire

Pour effectuer cette analyse, nous considérons à l'entrée du système un bruit blanc gaussien de moyenne nulle et de variance unité puis un signal USASI (bruit stationnaire corrélé similaire à l'enveloppe spectrale moyenne des signaux de parole) [1]. Le système à identifier est modélisé par une RIF à 1024 coefficients obtenue par troncature d'une réponse impulsionnelle acoustique (figure 3.2), mesurée expérimentalement dans une salle de dimension  $4 \times 4$  m. La variance du bruit ajouté à la sortie du filtre est choisie de sorte à avoir un RSB de 60 dB.

La figure 3.3 illustre l'influence du facteur de recouvrement  $\alpha$  sur la variation de l'EQM en fonction du nombre d'échantillons traités par l'algorithme  $GMDF\alpha$ . Nous avons considéré trois valeurs du facteur de recouvrement,  $\alpha = 1, 2$ , et  $4$ . La courbe du NLMS est également tracée pour comparaison. On constate dans un premier temps que (pour  $\alpha = 1$ ) le NLMS converge plus vite que le  $GMDF\alpha$ . Par accroissement du facteur de recouvrement, la vitesse de convergence devient équivalente ( $\alpha = 2$ ) et même supérieure ( $\alpha = 4$ ) à celle du NLMS. Par ailleurs, on remarque que le gain en vitesse de convergence est plus faible pour l'accroissement de 2 à 4 que de 1 à 2, ceci nous mène à déduire que l'utilisation d'un très grand facteur de recouvrement ( $\alpha > 4$ ) engendrerait un gain en vitesse de convergence encore moins significatif avec une complexité arithmétique accrue, ce qui n'est pas très avantageux.

La figure 3.4, montre les performances du NLMS et du GMDF pour un signal d'entrée corrélé (bruit USASI) et des facteurs de recouvrement  $\alpha = 1, 2$  et  $4$ . On constate que l'algorithme GMDF converge beaucoup plus vite que le NLMS. Ceci confirme l'avantage de la normalisation en fréquence dans le cas des signaux corrélés.

Le comportement de l'algorithme en fonction du pas d'adaptation est mis en évidence en examinant les résultats de la figure 3.5. On remarque que la variation du pas d'adaptation influe sensiblement sur la convergence de l'algorithme. Un pas trop faible induit une convergence trop lente, par contre un pas d'adaptation trop élevé accélère la convergence mais induit soit une EQMR trop élevée, soit une erreur initiale trop importante ou encore une divergence. On peut

conclure que le choix du pas d'adaptation doit prendre en considération trois paramètres importants : le temps de convergence, l'erreur quadratique résiduelle et la stabilité de l'algorithme.

La figure 3.6 montre la comparaison entre la structure contrainte et non contrainte de l'algorithme GMDF. On remarque que l'algorithme contraint converge moins rapidement mais avec une précision plus importante que le non contraint. Ceci s'explique par le fait que les termes de repliement, apparaissant dans la mise à jour non contrainte des coefficients du filtre, induisent des erreurs de précision qui affectent la convergence de l'algorithme.

L'influence de la segmentation de la réponse impulsionnelle sur la convergence de l'algorithme contraint est discutée (figure 3.7). On constate que la segmentation accélère la convergence. Ceci s'explique par le fait que l'augmentation du nombre de sous blocs réduit le nombre d'échantillons traités à chaque nouvelle itération et par conséquent l'adaptation des coefficients du filtre se fait plus fréquemment, d'où l'accélération de la convergence. Dans le cas de l'algorithme non contraint (figure 3.8) la segmentation ne présente pas d'influence notable sur la convergence de l'algorithme.

En faisant varier la variance du bruit blanc ajouté à la sortie du système, on peut discuter la robustesse de l'algorithme vis-à-vis du bruit. D'après les résultats obtenus (figure 3.9), la diminution du RSB induit une augmentation de l'EQMR. Ceci démontre que la présence d'un bruit additif indépendant du signal d'entrée et de forte puissance perturbe le niveau de convergence de l'algorithme.

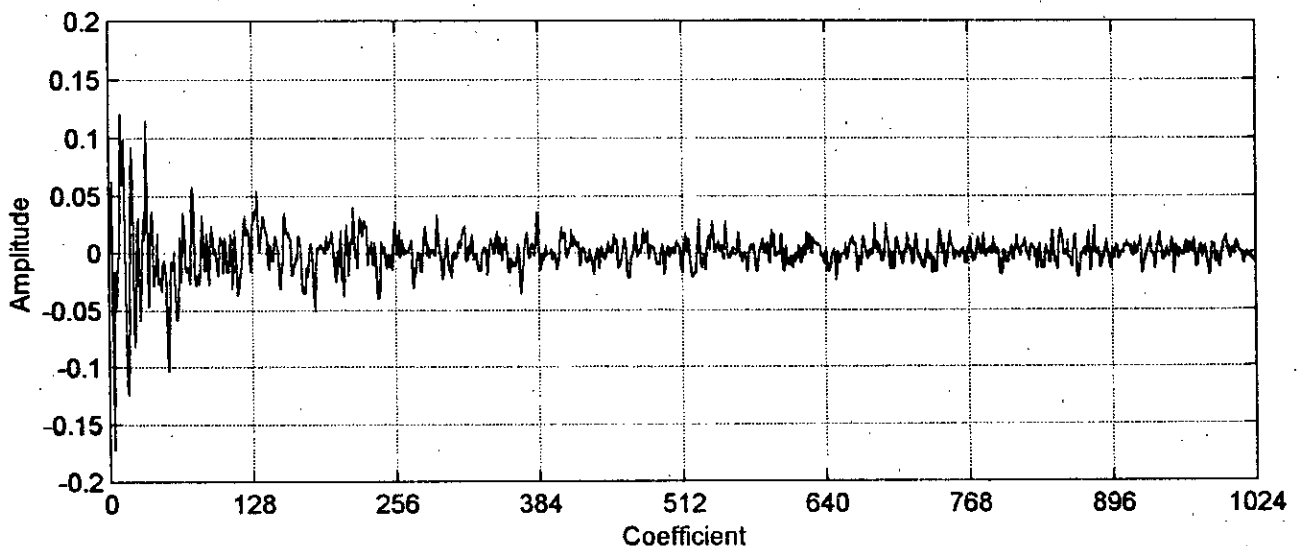


Figure 3.2 : Réponse impulsionnelle utilisée dans les simulations

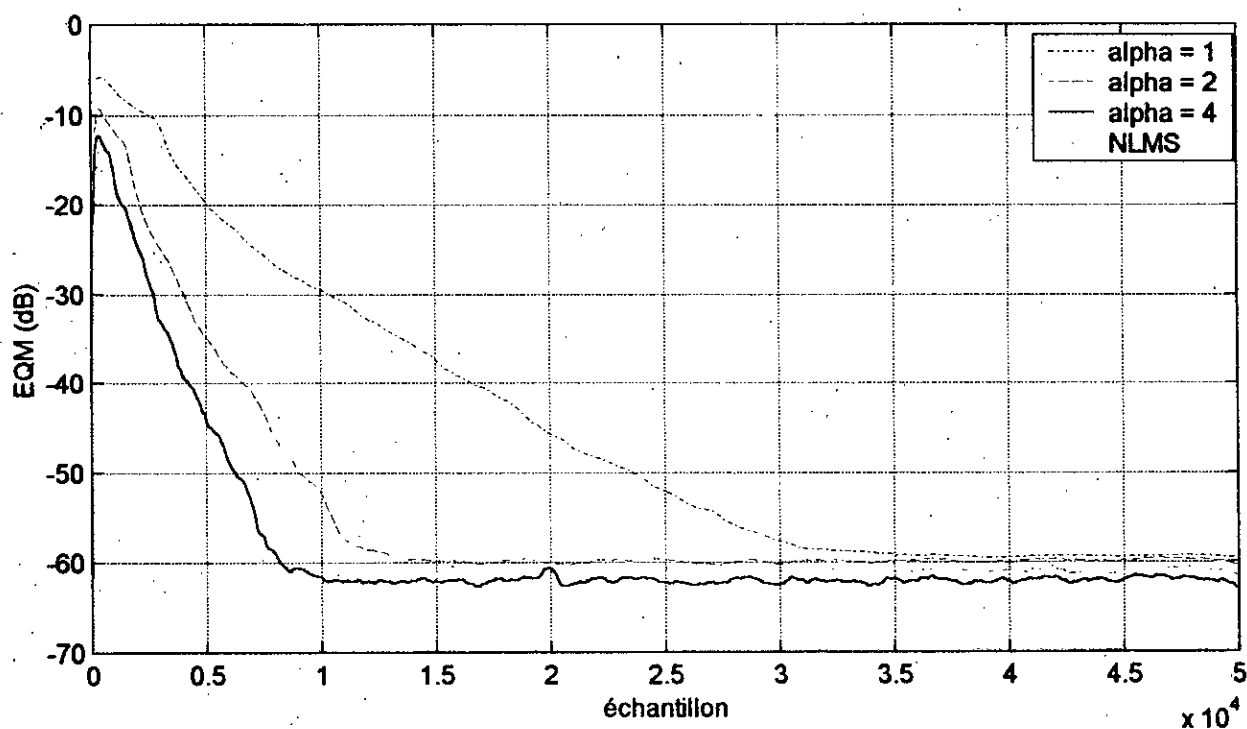


Figure 3.3 : Comparaison entre le NLMS et le GMDF contraint.

$L = 1024, K = 4, \mu = 0.06, \gamma = 0.92.$

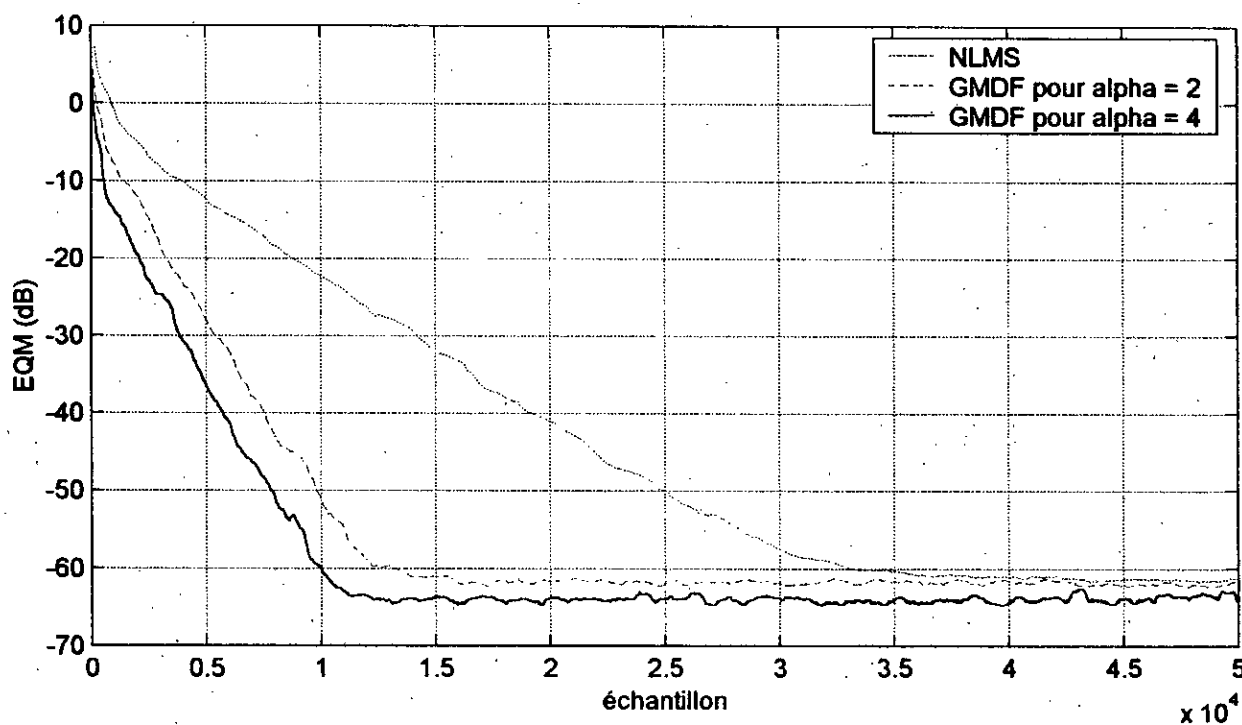


Figure 3.4 : Comparaison entre le NLMS et le GMDF contraint pour une entrée USASI.

$L = 1024, K = 4, \mu = 0.06, \gamma = 0.92.$



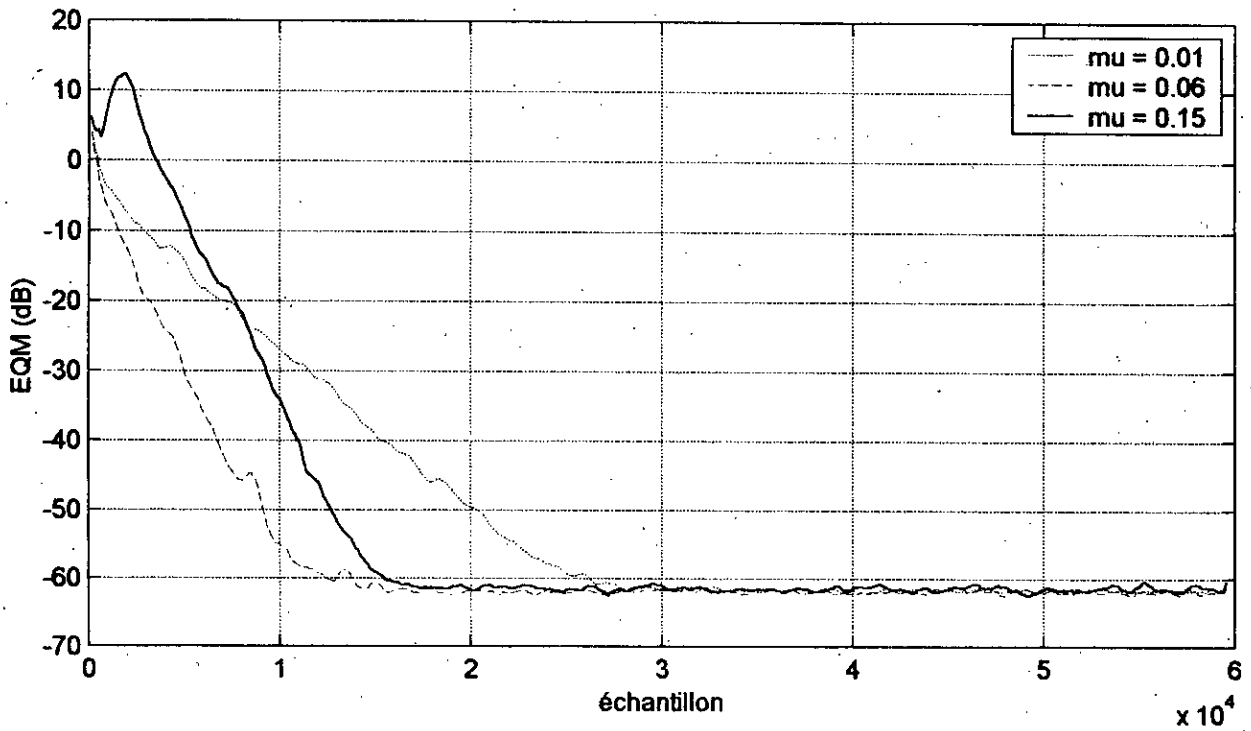


Figure 3.5 : Influence du pas d'adaptation sur la convergence du GMDF contraint.

$$L = 1024, K = 4, \alpha = 2, \gamma = 0.92.$$

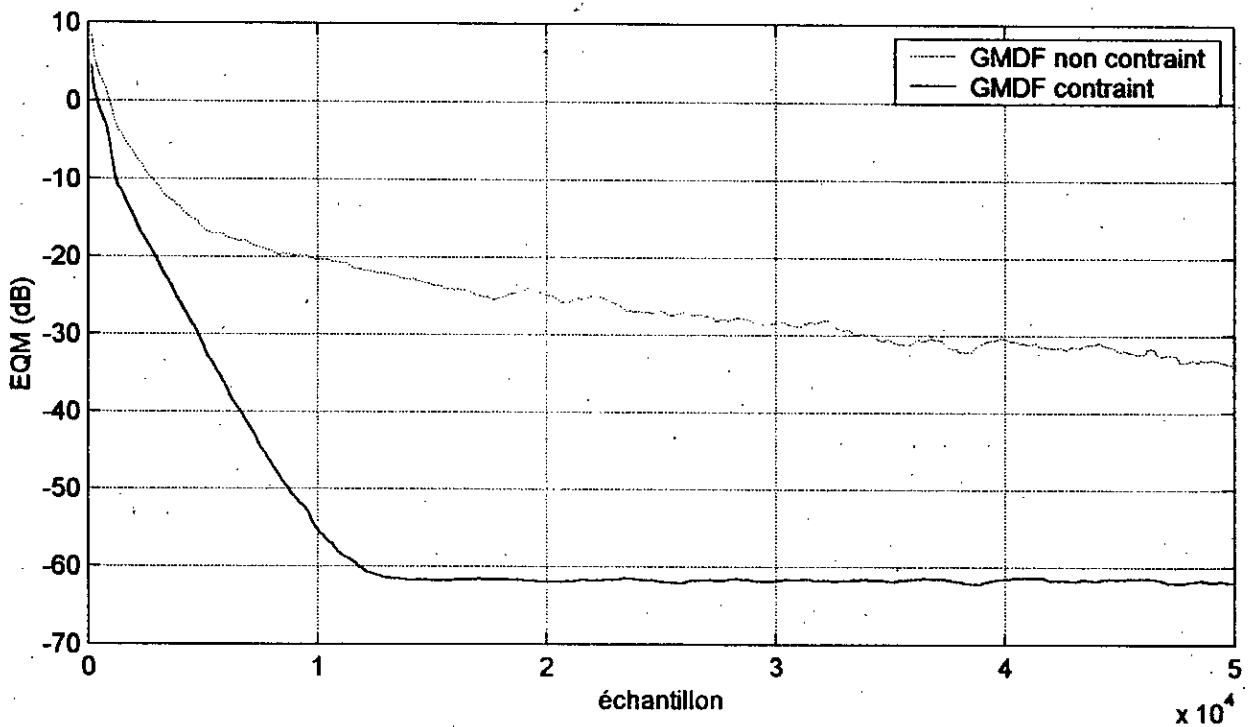


Figure 3.6 : Comparaison entre le GMDF contraint et le GMDF non contraint.

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$

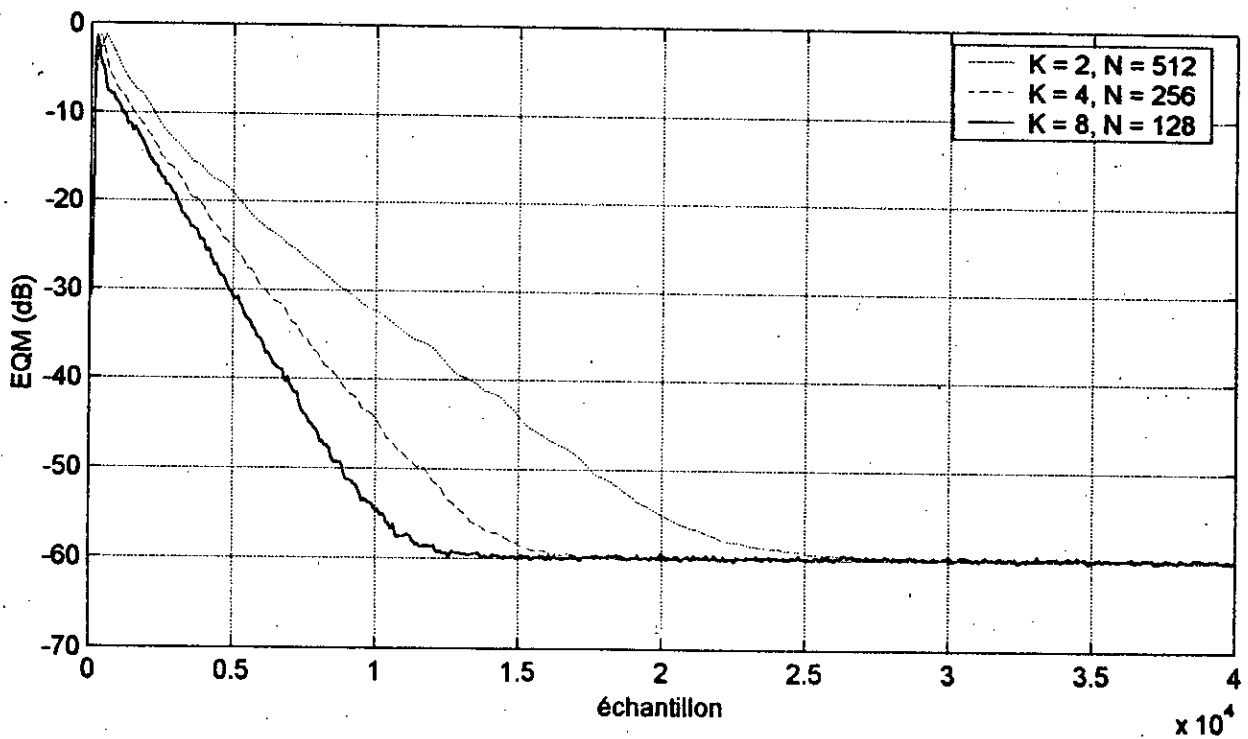


Figure 3.7 : Influence de la segmentation sur la convergence du GMDF contraint.

$L = 1024, \alpha = 2, \mu = 0.06, \gamma = 0.92.$

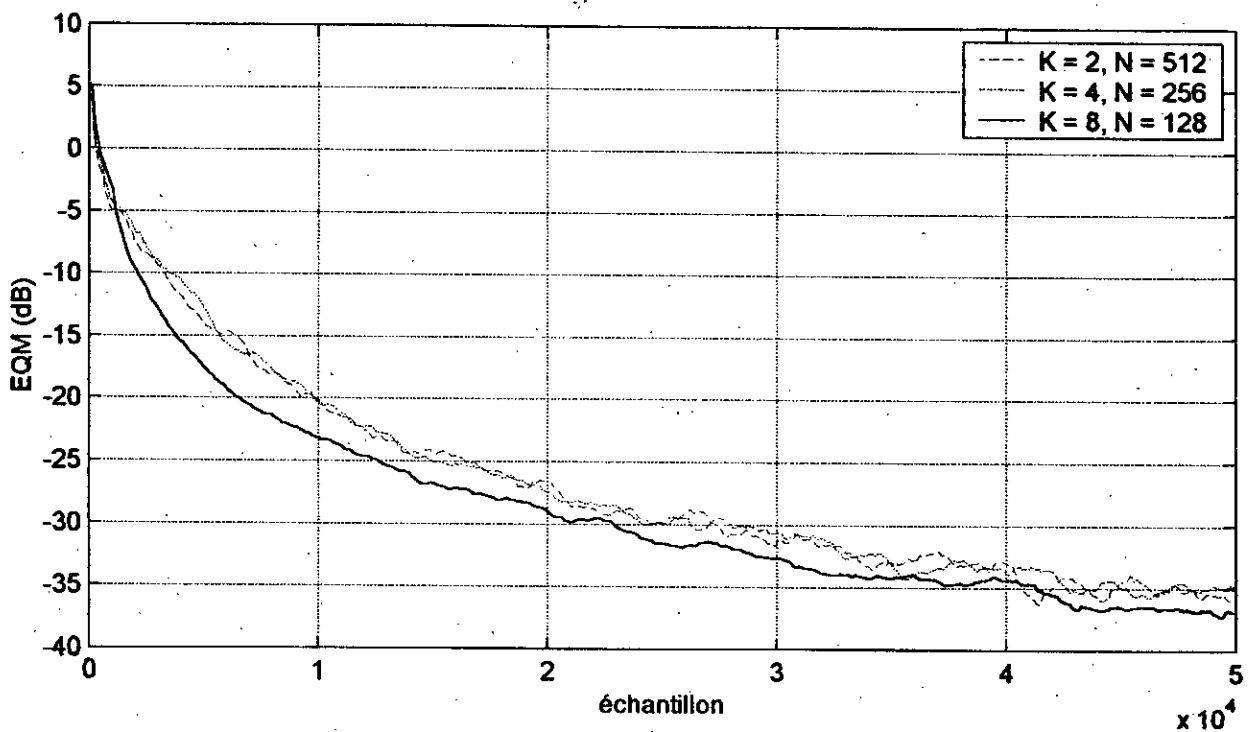


Figure 3.8 : Influence de la segmentation sur la convergence du GMDF non contraint.

$L = 1024, \alpha = 2, \mu = 0.06, \gamma = 0.92.$

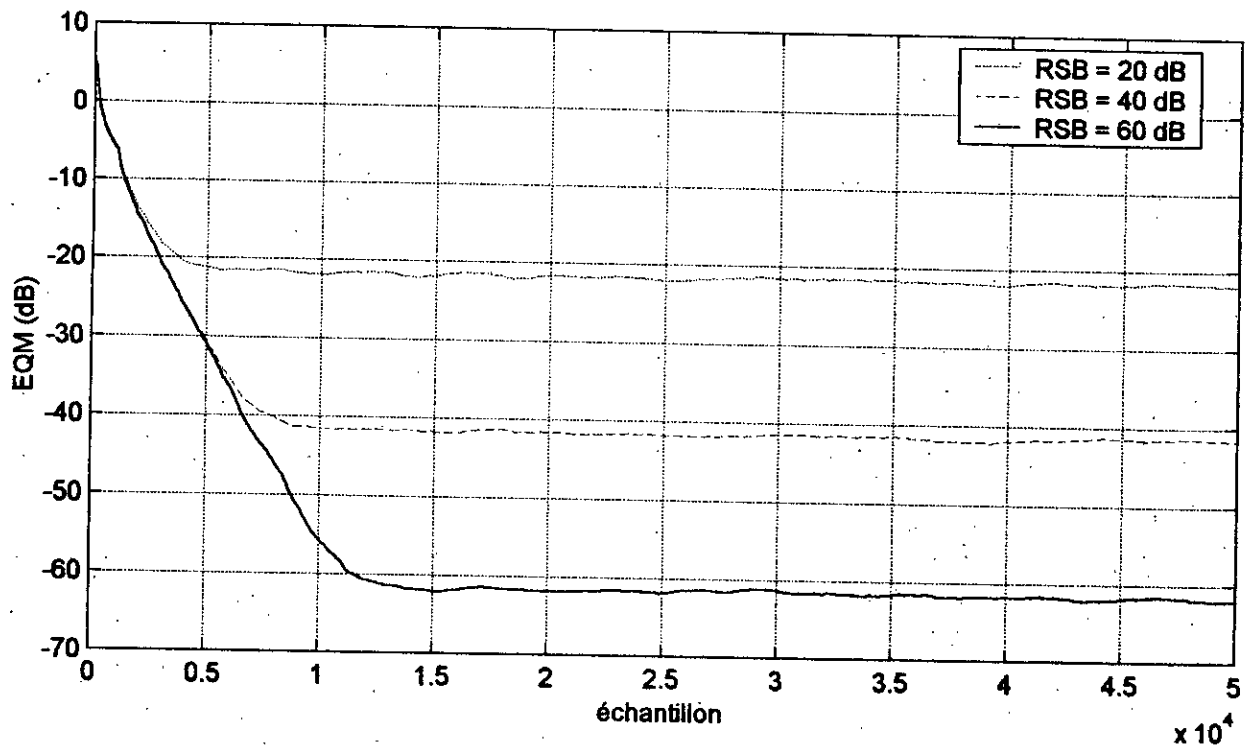


Figure 3.9 : Influence du bruit additif sur la convergence du GMDF contraint

$$L = 1024, K = 2, \alpha = 2, \mu = 0.05, \gamma = 0.92$$

### 3.4 Analyse dans un contexte non stationnaire

Dans cette étape nous allons considérer la même réponse impulsionnelle que celle utilisée dans l'étape précédente. La non-stationnarité est introduite soit par les signaux d'entrée ou par une variation du système à identifier.

#### 3.4.1 Analyse avec des signaux de parole

Les signaux de parole utilisés dans cette analyse ont été enregistrés sur une durée de 10 s et échantillonnés à une fréquence de 11 KHz. Les simulations ont été effectuées pour tester l'applicabilité de l'algorithme GMDF $\alpha$  à l'annulation d'écho acoustique où les signaux à traiter sont corrélés et non stationnaires. Pour cela, on considère trois situations :

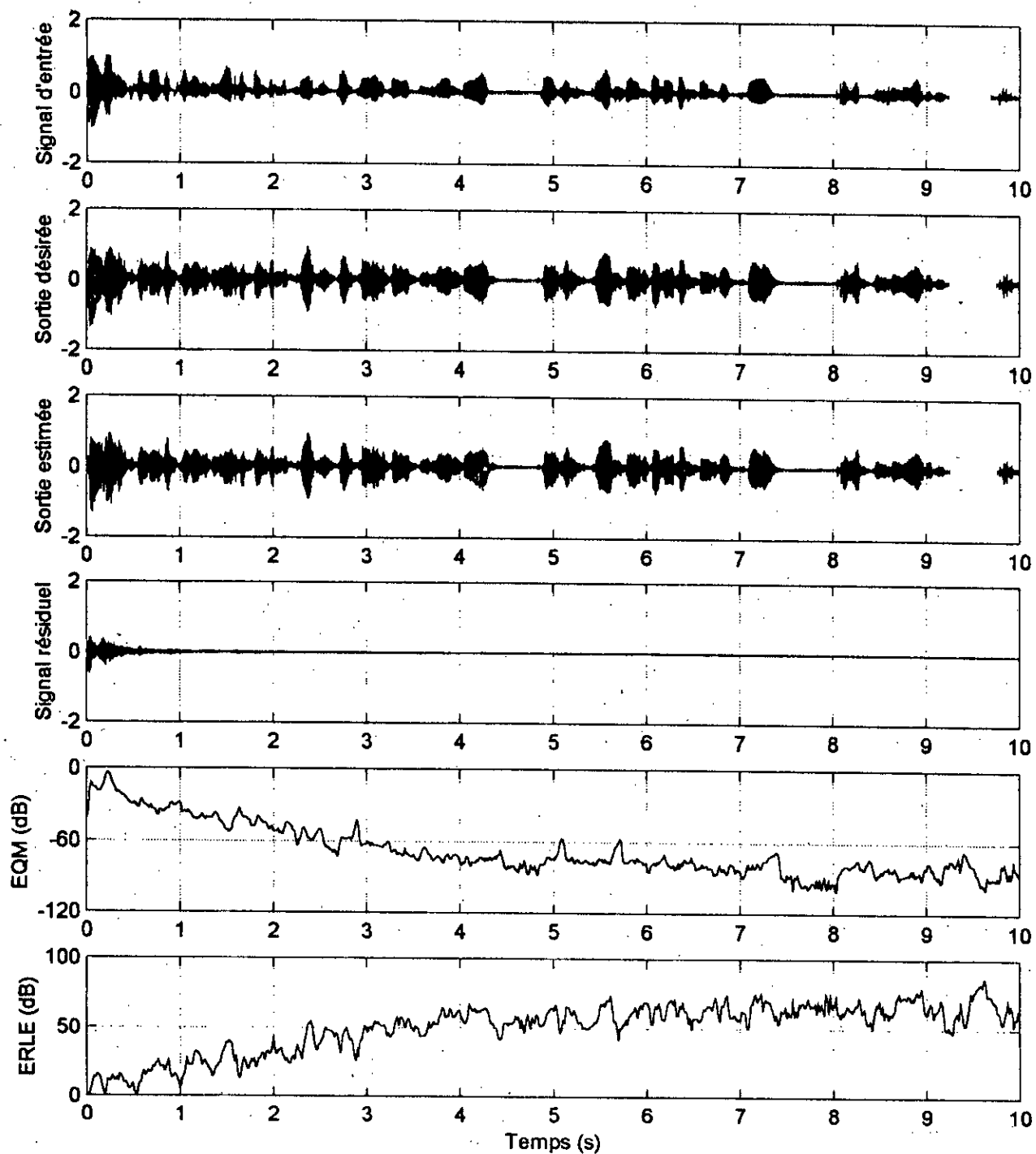
- La sortie désirée est constituée uniquement de la convolution entre le signal d'entrée et la réponse impulsionnelle du système à identifier.
- On superpose à la sortie du filtre un bruit blanc gaussien de sorte à avoir un RSB variable.
- On superpose à la sortie du filtre un deuxième signal de parole (Locuteur B).

La figure 3.10 montre les résultats de la simulation du premier cas. On constate que l'erreur d'estimation (signal résiduel) est pratiquement nulle après convergence avec un ERLE de l'ordre de 80 dB. La figure 3.11 montre une comparaison entre les performances du GMDF et du NLMS dans cette situation. On constate que l'algorithme GMDF atteint un niveau de performances nettement supérieure au NLMS.

Le figure 3.12 montre les résultats de la simulation dans le cas où un bruit blanc gaussien de moyenne nulle est superposé à la sortie du système. On constate que le signal récupéré à la sortie du filtre adaptatif (signal résiduel) correspond au bruit ajouté. Par augmentation de la puissance du bruit on constate que ce résultat n'est valable que pour une faible puissance du bruit comparées à celle de la sortie du système.

Les résultats obtenus dans la situation précédente nous mène à remplacer le bruit par un deuxième signal de parole (figure 3.13). On constate que le signal résiduel correspond au deuxième signal de parole perturbé par un petit signal résiduel dû à une erreur d'identification de la réponse impulsionnelle du système. Ceci nous mène à conclure cette partie des simulations en retenant que la perturbation du filtre adaptatif par un autre signal influe sur la convergence d'autant plus que la puissance du signal perturbateur devient importante.

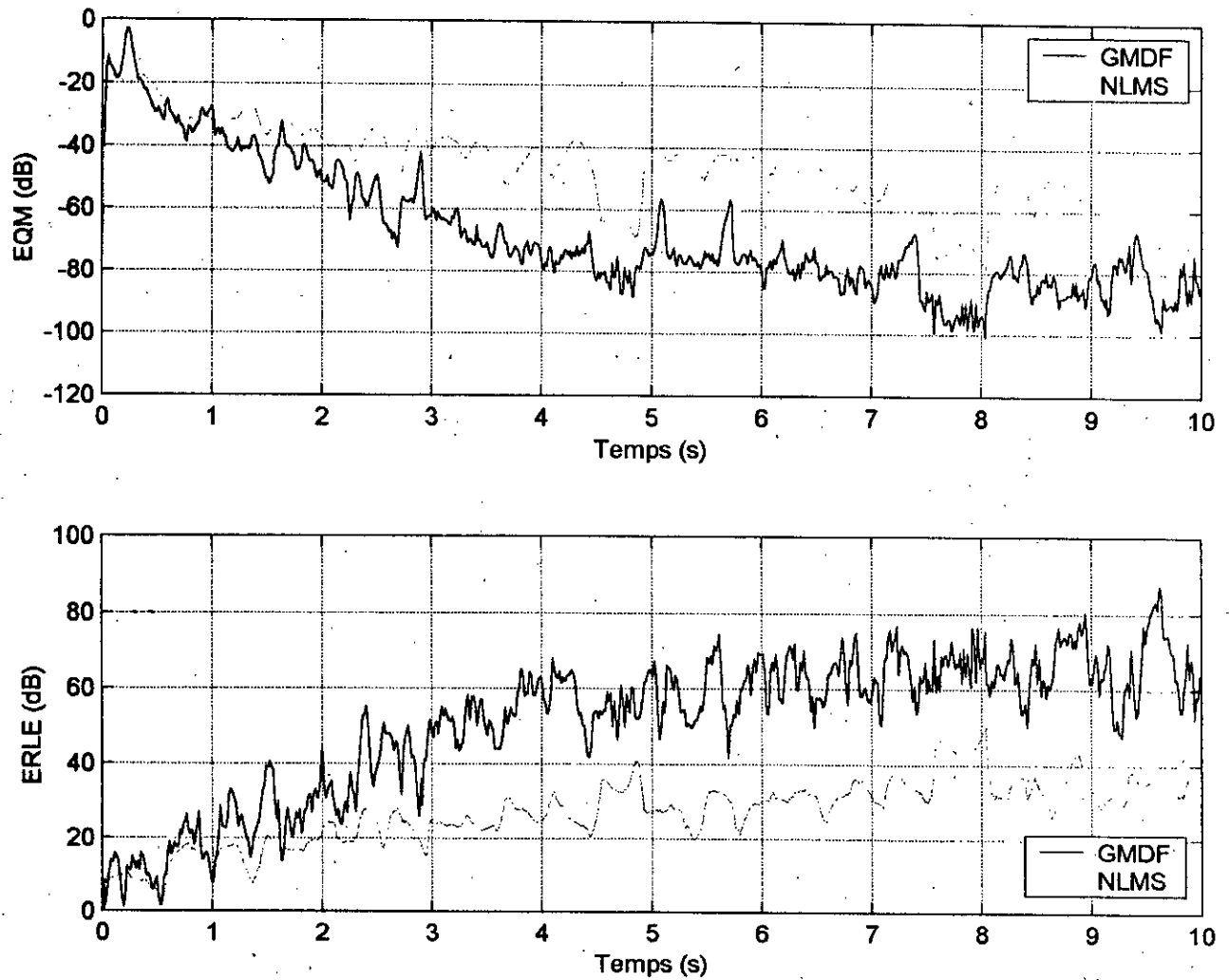
L'application de l'algorithme  $\text{GMDF}_\alpha$  non contraint dans cette situation (figure 3.14) montre que celui-ci peut être efficacement appliqué mais avec des performances inférieures par rapport au  $\text{GMDF}_\alpha$  contraint.



**Figure 3.10** : Annulation d'un écho acoustique non perturbé

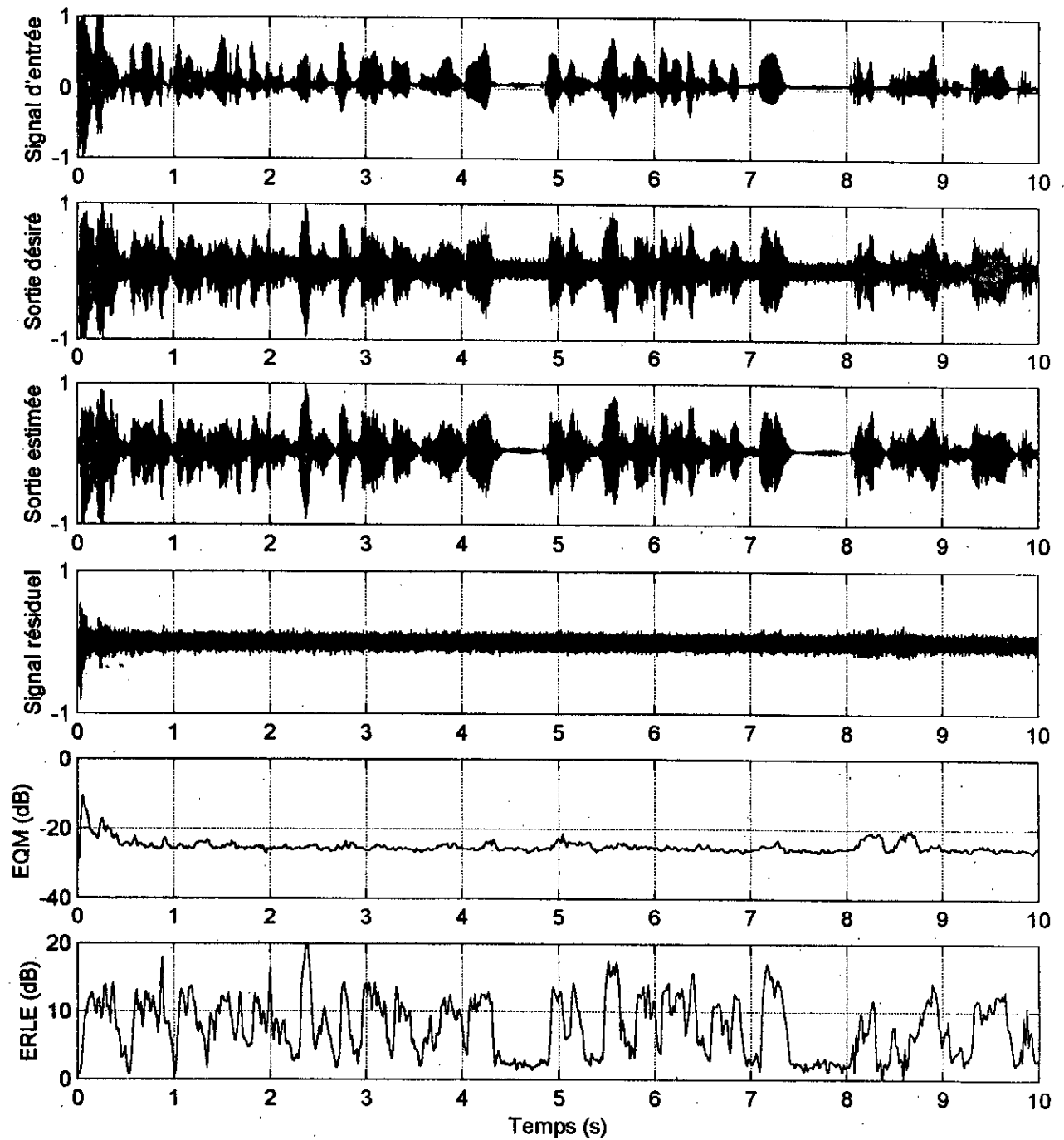
Algorithme GMDF $\alpha$  contraint

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$



**Figure 3.11** : Comparaison entre le NLMS et le GMDF pour un signal d'entrée non stationnaire

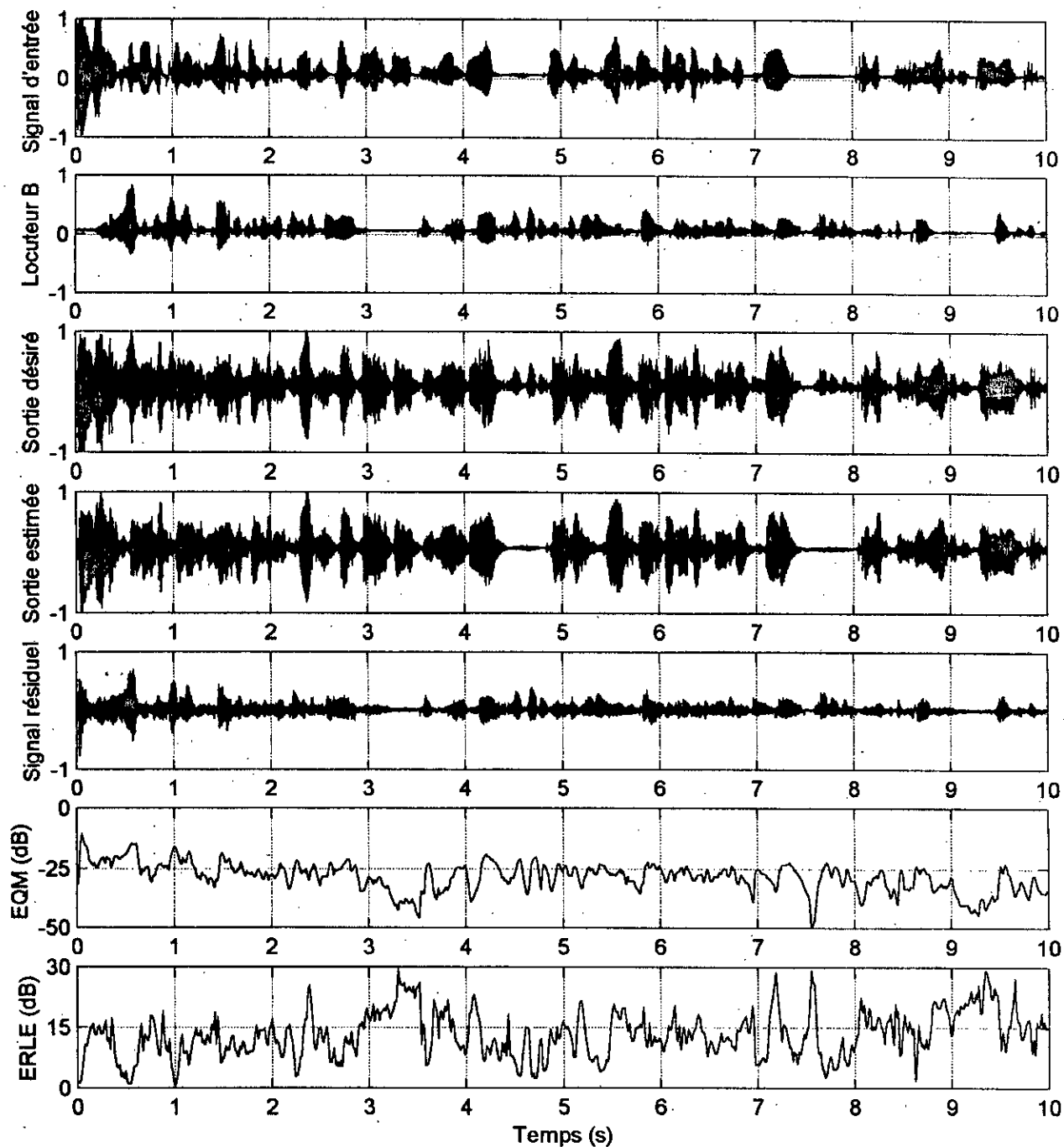
$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$



**Figure 3.12 :** Annulation d'un écho acoustique perturbé par un bruit additif

Algorithme  $GMD\alpha$  contraint

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92, RSB = 10 \text{ dB}$$

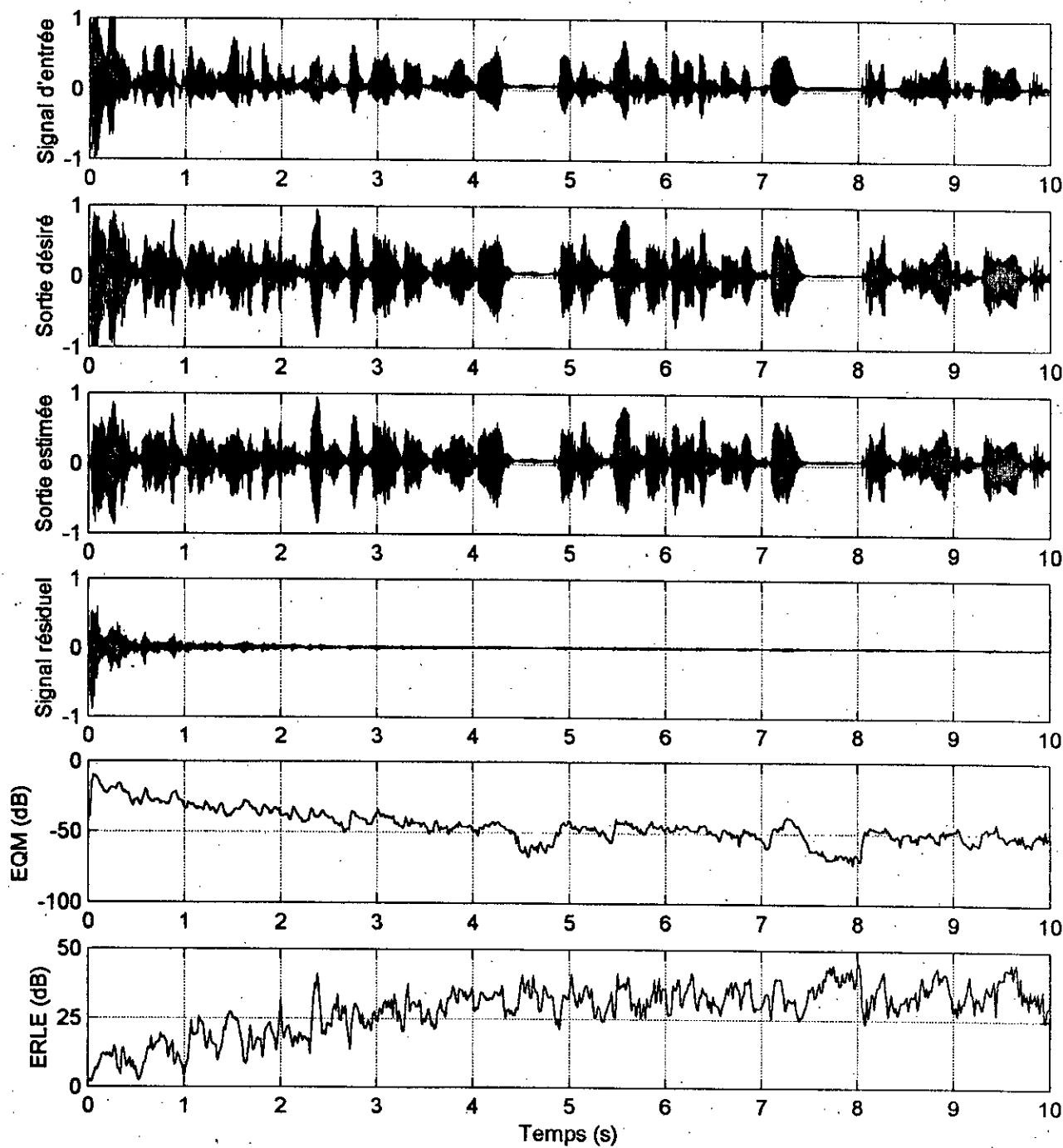


**Figure 3.13 :** Annulation d'un écho acoustique perturbé par un signal de parole

*Algorithme GMDF $\alpha$  contraint*

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$





**Figure 3.14 :** *Annulation d'un écho acoustique non perturbé*

*Algorithme GMDF $\alpha$  non contraint*

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$

### 3.4.2 Effet d'une variation du système

Nous allons maintenant étudier la capacité de poursuite de l'algorithme en imposant une variation linéaire de la réponse impulsionnelle du système à identifier suivant la relation [24] :

$$h_k(n+1) = h_k(n) + \varepsilon |h_k(n)| b_k(n) \quad \text{pour } k=0, 1, \dots, L-1$$

où  $h_k(n)$  est le  $k^{\text{ème}}$  coefficient de la réponse impulsionnelle à l'instant  $n$ ,  $\varepsilon$  une constante de faible valeur et  $b_k(n)$  un bruit blanc gaussien de moyenne nulle et variance unité. On maintient la réponse impulsionnelle constante entre les instants  $t = 0$  s et  $t_0 = 3$  s. La variation linéaire est introduite suivant la relation précédente entre les instants  $t_0$  et  $t_1 = 4.5$  s. A partir de  $t_1$  on fige la réponse impulsionnelle.

Les résultats de cette simulation, pour  $\varepsilon = 0.05$  et pour une entrée USASI sont montrés dans la figure 3.15, dans laquelle on discute la variation l'EQM pour différentes valeurs de  $\alpha$ . Le résultat de l'utilisation du NLMS est également tracé pour comparaison. On observe bien l'amélioration de la capacité de poursuite par l'accroissement du facteur de recouvrement. On constate également un meilleur comportement par rapport au NLMS.

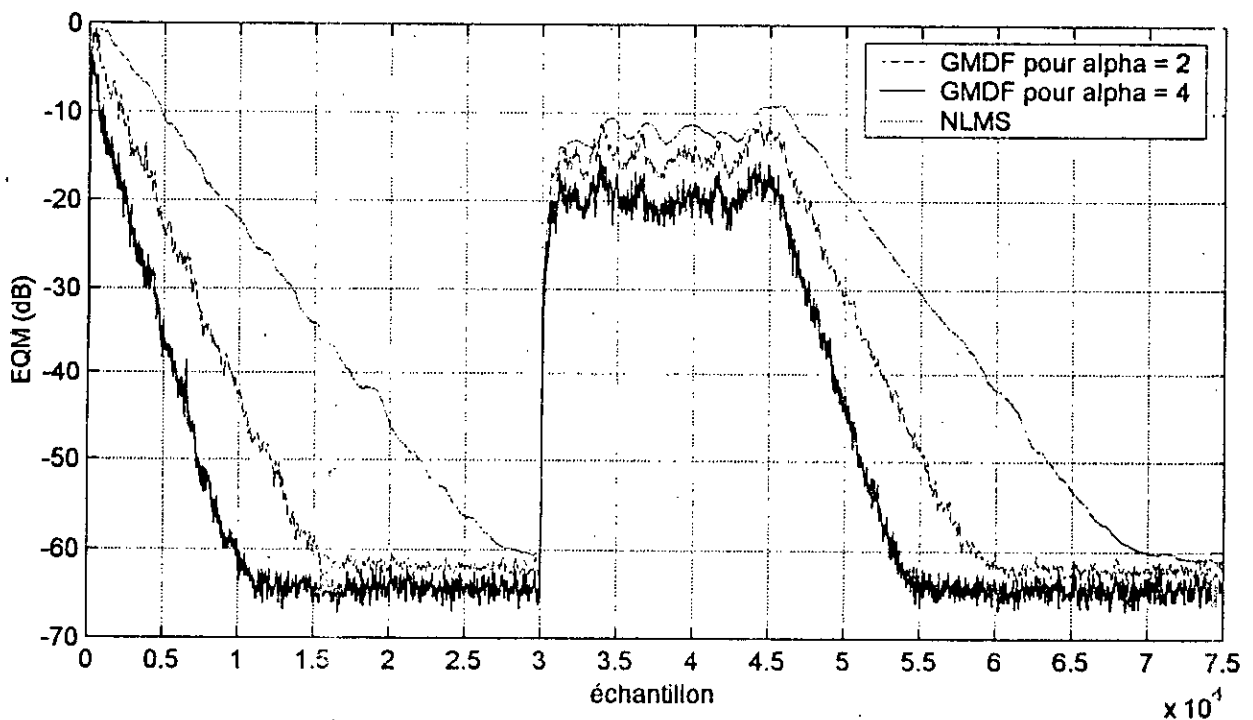


Figure 3.15 : Effet d'une variation du système à identifier.

$$L = 1024, K = 4, \mu = 0.06, \gamma = 0.92.$$

### 3.5 Conclusion

Dans ce chapitre nous avons présenté les résultats des différentes simulations effectuées dans plusieurs contextes. L'analyse des performances nous a montré que l'algorithme GMDF $\alpha$  constitue un outil très efficace pour l'identification des systèmes modélisés par de très longues réponses impulsionnelles. Une des caractéristiques essentielles de cet algorithme réside dans le fait qu'il permet un contrôle du délai de traitement par la segmentation de la réponse impulsionnelle d'une part, d'autre part, l'introduction du facteur de suréchantillonnage offre une mise à jour plus fréquente des coefficients du filtre, ce qui accélère la vitesse de convergence et améliore la capacité de poursuite.

Les analyses effectuées dans des situations non stationnaires ont montré que cet algorithme de filtrage adaptatif présente une structure avantageuse en vue de son application à l'annulation d'écho acoustique dans un environnement réel.

# Chapitre 4

## Application avec des signaux expérimentaux

Nous présenterons dans ce chapitre les résultats expérimentaux de l'application de l'algorithme  $\text{GMDF}\alpha$  à l'annulation d'écho acoustique dans une situation de téléconférence. L'expérimentation a été réalisée à l'aide de la carte son d'un micro-ordinateur, d'un microphone et d'un haut-parleur.

### 4.1 Notations

Par souci de clarté, nous rappelons ci-dessous (figure 4.1) le schéma de principe d'une communication en téléconférence avec annulation d'écho acoustique, ainsi que les définitions des signaux et les notations que nous utiliserons dans les descriptions qui suivront.

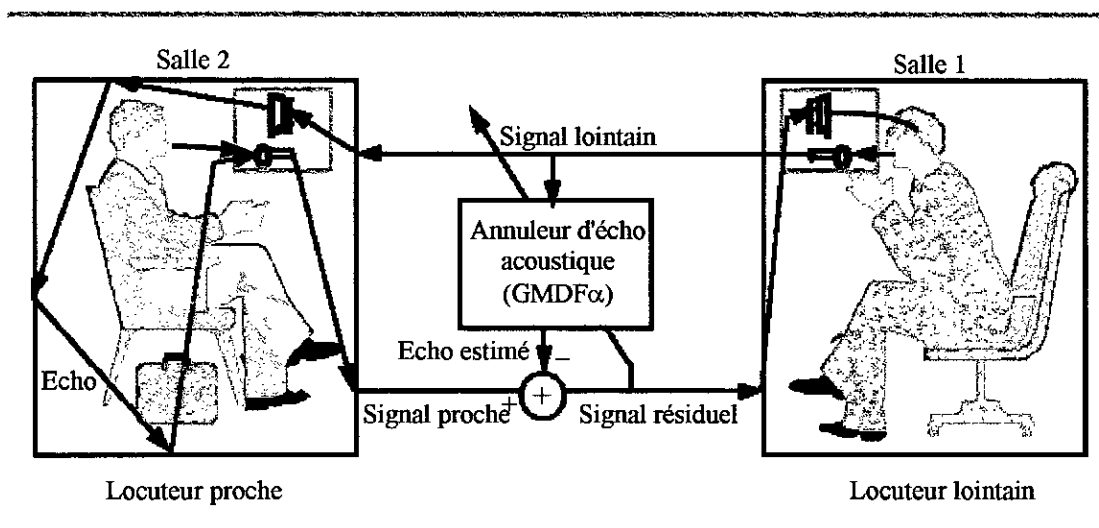


Figure 4.1: Principe de Téléconférence avec annulation d'écho acoustique

Le signal provenant de la première salle est appelé signal lointain et celui de la deuxième salle, signal proche. Le signal proche est constitué par l'écho et le signal perturbateur qui est

composé du signal du locuteur proche et du bruit ambiant dans la deuxième salle. A la sortie de l'annuleur d'écho on retrouve l'écho estimé qui sera soustrait du signal proche afin de produire le signal résiduel.

## 4.2 Mécanisme expérimental d'acquisition des signaux

Afin de nous placer dans une situation semblable à la téléconférence, l'enregistrement des signaux a été effectué en adoptant les démarches suivantes :

- (1) On effectue un premier enregistrement d'un signal de parole d'une durée de 10 secondes échantillonné à une fréquence de 11025 Hz. Ce premier signal sera considéré comme étant celui du locuteur lointain. On signale dans cette étape que le choix de la fréquence d'échantillonnage est fait de sorte à assurer une compatibilité avec les fréquences admises par la carte audio utilisée.
- (2) On génère la séquence enregistrée précédemment via le haut-parleur et on effectue en même temps un deuxième enregistrement par le microphone, placé à une distance de 1m du haut-parleur. Ce deuxième enregistrement correspond à l'écho généré par la première séquence. On note que cette démarche correspond exactement à la situation de téléconférence, dans laquelle le signal du locuteur lointain (enregistrement 1) est reçu via le haut-parleur et dont une partie se réinjecte dans le microphone (enregistrement 2). Dans cette étape deux cas se présentent : l'absence du signal d'un deuxième locuteur nous mène vers la situation dite de simple duplex, par contre la présence d'un deuxième locuteur est une situation de full duplex.
- (3) En utilisant l'annuleur d'écho basé sur l'algorithme GMDF $\alpha$ , on effectue l'identification de la réponse impulsionnelle modélisant la salle et par conséquent une estimée de l'écho peut être générée afin d'annuler celui-ci.

## 4.3 Résultats expérimentaux

Afin de nous placer dans des situations semblables à celles rencontrées en téléconférence, nous avons effectué les expérimentations en quatre étapes : situation de simple duplex, full duplex et enfin full duplex en présence de bruit ambiant.

### 4.3.1 Application dans le cas "simple duplex"

Dans cette situation, le signal proche est constitué uniquement de l'écho et d'un signal de bruit de faible puissance (bruit de fond et celui de l'environnement de la salle). On discutera dans cette expérience l'influence de la taille de la réponse impulsionnelle sur le taux de d'annulation de l'écho (ERLE).

Les figures 4.2 et 4.3 montrent les résultats de l'application de l'algorithme  $\text{GMDF}\alpha$  contraint à l'annulation d'écho acoustique en utilisant une réponse impulsionnelle de 1024 coefficients. On constate que les 80 premiers coefficients de la réponse impulsionnelle estimée sont nuls. Ceci traduit le délai de propagation du signal entre le haut-parleur et le microphone suivant le chemin direct (il faut prendre compte également du retard introduit par l'instruction MATLAB du début de génération et d'enregistrement de l'écho), d'autre part les autres coefficients correspondent aux différentes trajectoires suivies par le signal d'écho.

La figure 4.4 montre la variation de l'EQM et de l'ERLE pour  $L = 512$  et 2048. On constate que l'augmentation de la taille de la réponse impulsionnelle induit une amélioration des performances (augmentation de l'ERLE de 8 dB et diminution de l'EQM de 8 dB). Par conséquent, on peut dire que la taille de la réponse impulsionnelle doit être suffisamment grande pour arriver à obtenir des performances acceptables. D'autre part une réponse impulsionnelle trop longue (au delà d'une certaine valeur) induirait inutilement une charge de calcul supplémentaire. Par conséquent un compromis devra être établi selon les performances souhaitées.

La figure 4.5 montre que l'algorithme  $\text{GMDF}\alpha$  non contraint peut être efficacement appliqué à l'annulation d'écho acoustique.

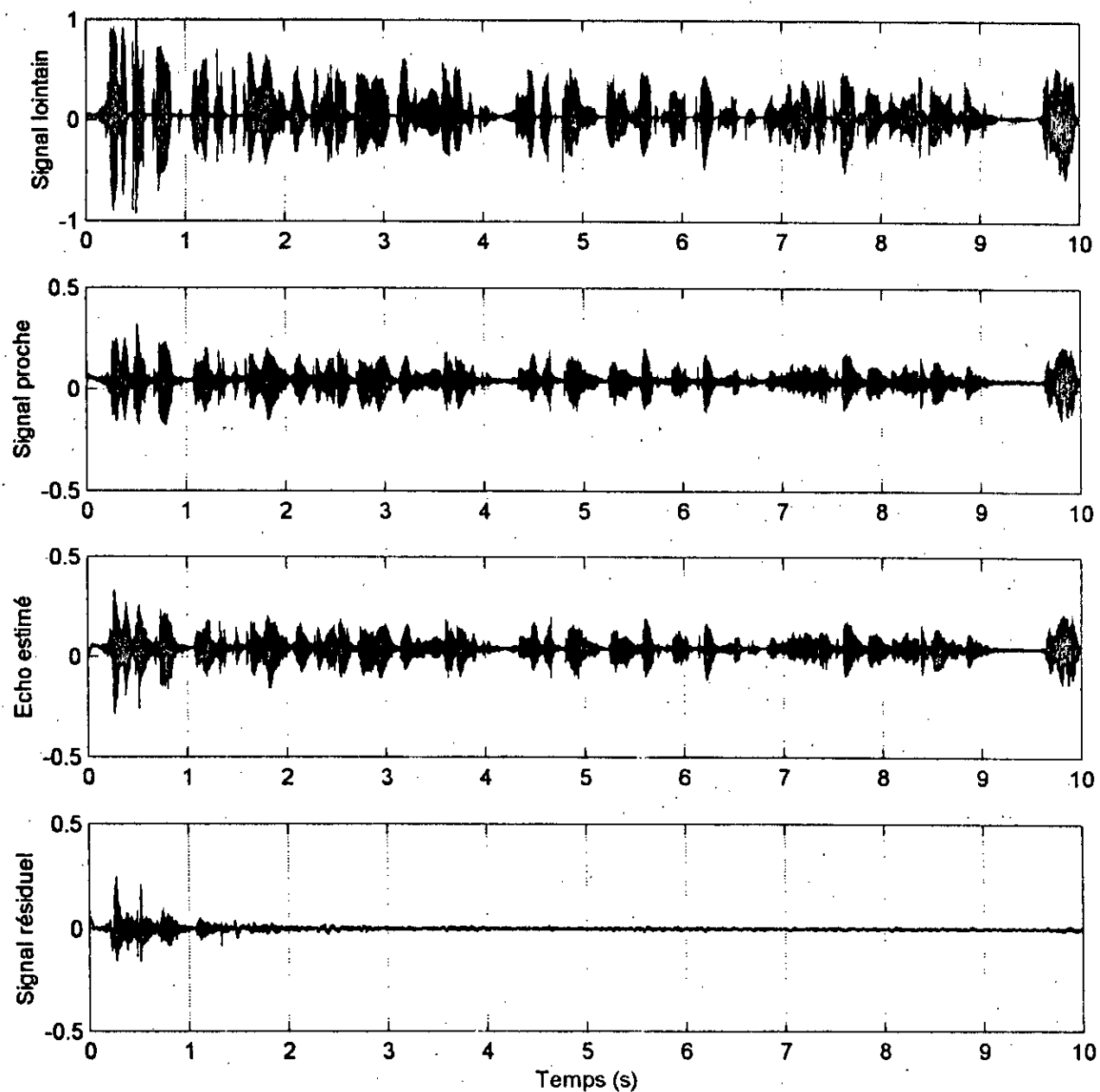


Figure 4.2 : Annulation d'écho acoustique dans le cas "Simple duplex"

Algorithme  $GMD\alpha$  contraint

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$

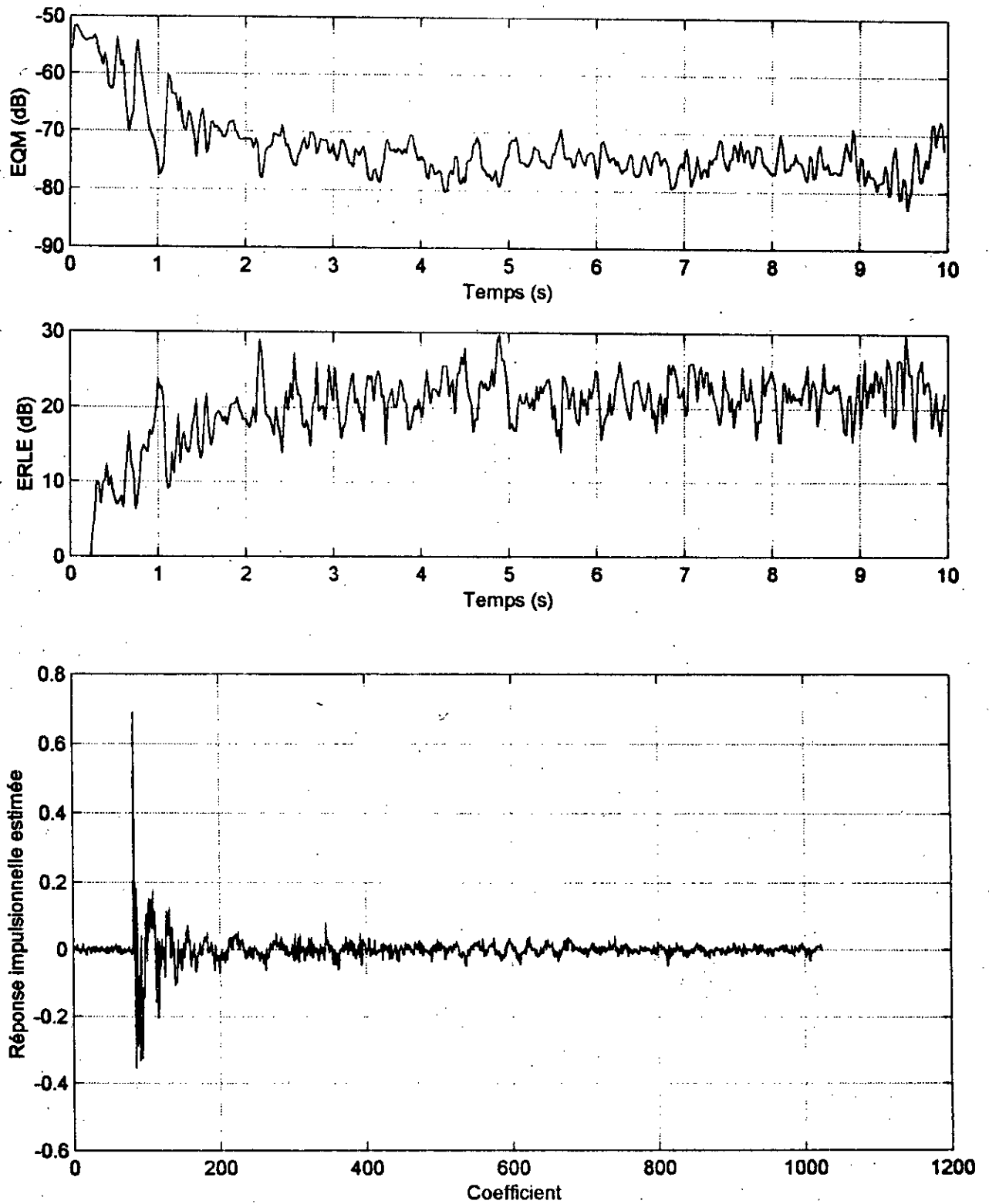


Figure 4.3 : Annulation d'écho acoustique dans le cas "Simple duplex"

Algorithme GMDF $\alpha$  contraint

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$



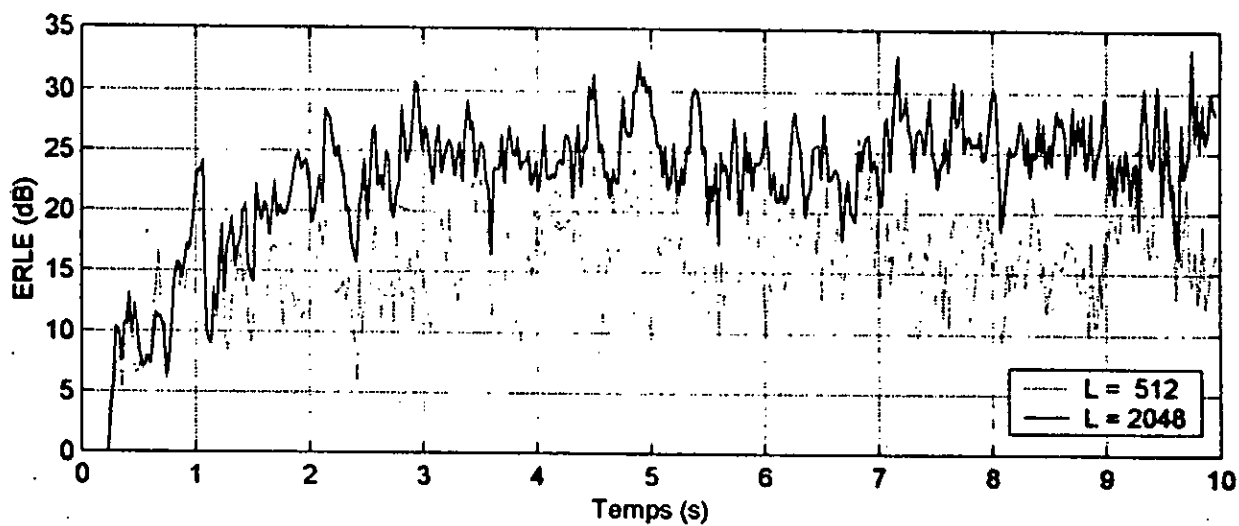
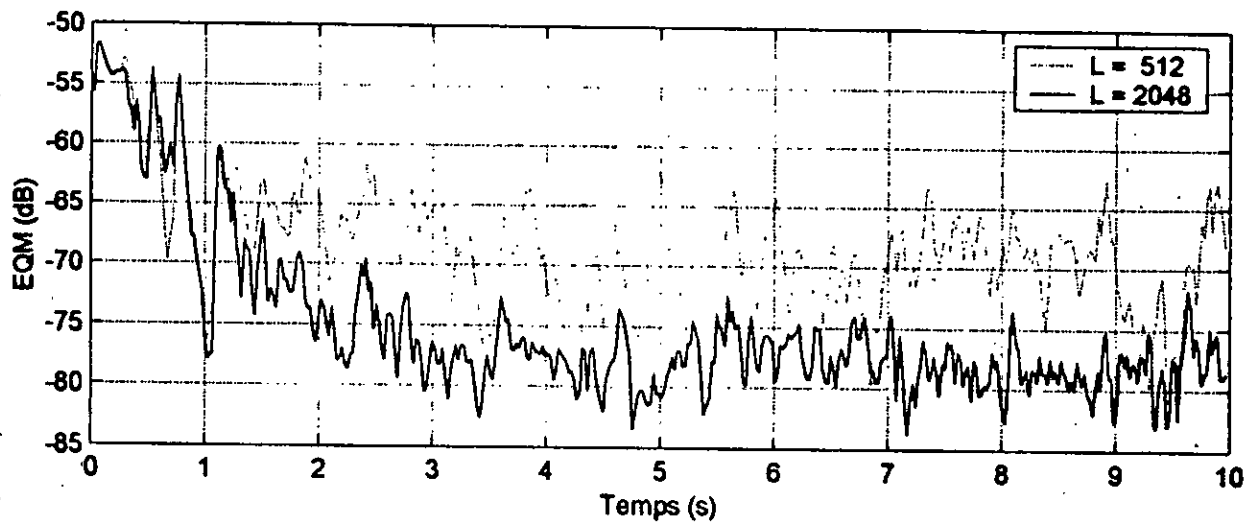


Figure 4.4 : Influence de la taille de la réponse impulsionnelle sur l'annulation d'écho

Algorithme GMDF $\alpha$  contraint

$K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$

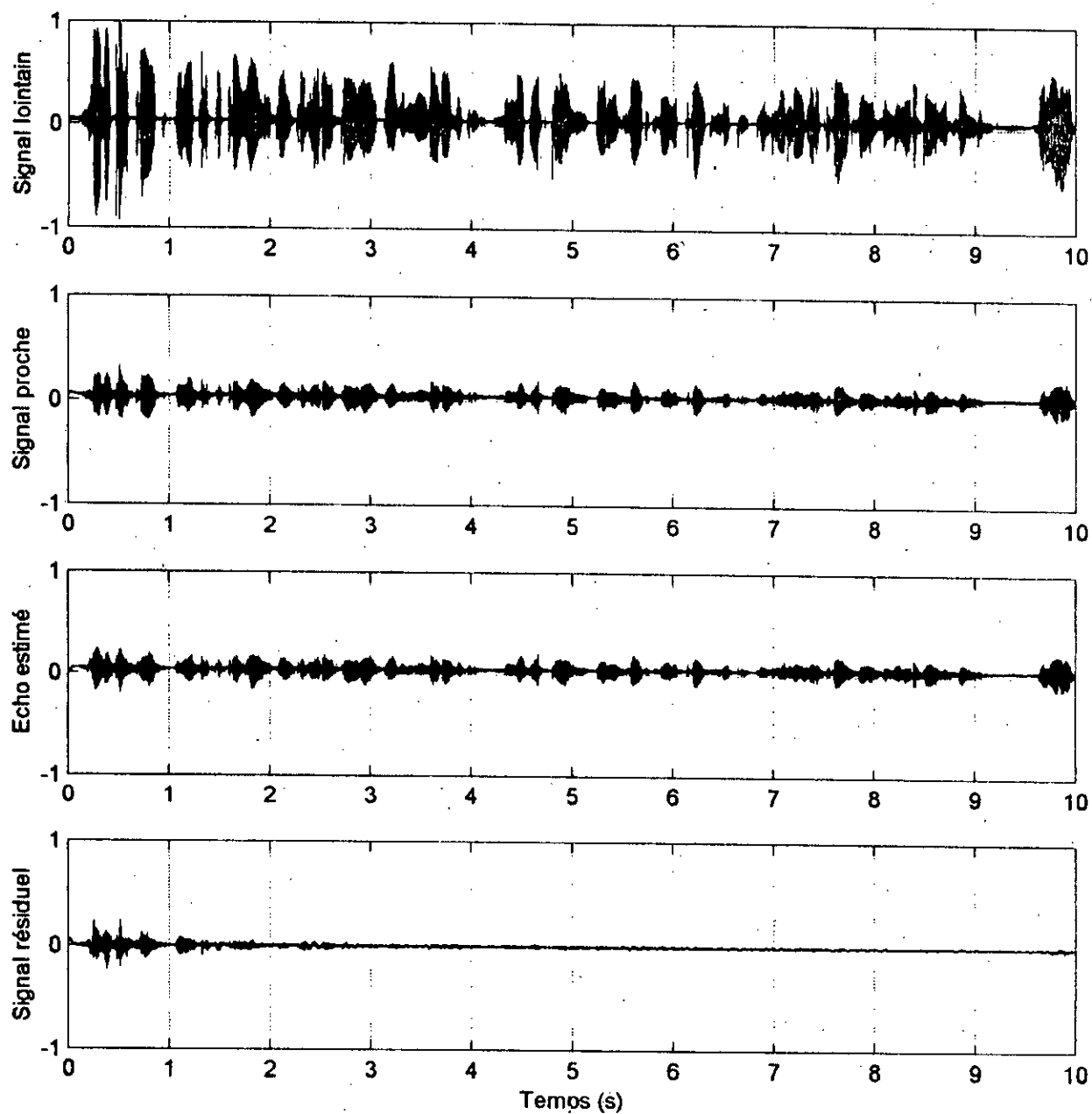


Figure 4.5 : Annulation d'écho acoustique dans le cas "Simple duplex"

Algorithme GMDF $\alpha$  non contraint

$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$

### 4.3.2 Application dans le cas "full duplex"

Les résultats de cette application (figure 4.6) montrent que la présence du second locuteur perturbe l'opération d'identification de la réponse impulsionnelle. Par conséquent l'annulation d'écho devient dégradée à cause des erreurs induites par le signal correspondant au second locuteur. Cette application nécessite donc le rajout d'une procédure de détection de la présence simultanée des signaux correspondants aux deux locuteurs. En présence du signal du second locuteur, les coefficients du filtre doivent être maintenus constants (pour éviter la divergence de l'algorithme si l'on effectue la mise à jour) et seule l'opération de filtrage sera effectuée.

Dans les situations pratiques cette opération (appelée *détection du full duplex*) est réalisée en mesurant la distance entre l'enveloppe spectrale du signal émis par le haut-parleur et celle du signal reçu par le microphone [23]. Une petite distance entre les deux enveloppes indique l'absence du signal du second locuteur. En revanche, une grande distance indiquerait la situation du full duplex. Ainsi une distance seuil doit être soigneusement fixée pour éviter des fausses détections.

En annulation d'écho acoustique, les signaux traités sont des signaux de parole, alors leur enveloppe spectrale peut être modélisée par un modèle autoregressif (AR) de faible ordre. Une première étape consiste à estimer le vecteur  $\mathbf{a}^{(s)}$  des coefficients du modèle AR d'ordre  $P$  correspondant au bloc courant (de taille  $M$ ) du signal lointain, en utilisant la méthode d'auto-corrélation [23]. En utilisant ces coefficients on calcule l'erreur résiduelle sur le bloc courant :

$$\varepsilon_n^{(s)} = x_{sR+n} + \sum_{k=1}^P a_k^{(s)} x_{sR+n-k} \quad \text{pour } n = 0, \dots, M \quad (4.1)$$

De façon similaire, l'erreur résiduelle sur le bloc du signal proche est aussi calculée en filtrant celui-ci par le modèle AR correspondant au signal lointain :

$$\eta_n^{(s)} = d_{sR+n} + \sum_{k=1}^P a_k^{(s)} d_{sR+n-k} \quad \text{pour } n = 0, \dots, M \quad (4.2)$$

La distance entre les deux enveloppes peut être représentée par le rapport entre les puissances des deux erreurs :

$$\rho^{(s)} = \frac{\sum_{n=0}^M (\eta_n^{(s)})^2}{\sum_{n=0}^M (\varepsilon_n^{(s)})^2} \quad (4.3)$$

L'idée de base est de comparer cette distance à une constante seuil variable  $\rho_s(s)$  mise à jour à chaque itération pendant l'absence du second locuteur.

L'adaptation de la constante seuil se fait après l'estimation de la valeur moyenne  $\rho_{\text{moy}}(s)$  et de la valeur maximale  $\rho_{\text{max}}(s)$  de  $\rho_s(s)$  en utilisant les récurrences suivantes [23] :

$$\rho_{\text{moy}}(s) = \lambda_1 \rho_{\text{moy}}(s-1) + (1-\lambda_1) \rho(s) \quad (4.4)$$

$$\rho_{\text{max}}(s) = \max\{\rho(s), \lambda_2 \rho_{\text{max}}(s-1)\} \quad (4.5)$$

où  $\lambda_1$  et  $\lambda_2$  sont des constantes de lissage comprises dans l'intervalle  $[0,1]$ .

La constante de seuil est alors mise à jour par :

$$\rho_s(s) = \rho_{\text{moy}}(s) + \lambda_3 (\rho_{\text{max}}(s) - \rho_{\text{moy}}(s)) \quad (4.6)$$

où  $\lambda_3$  est une constante qui peut être identique à  $\lambda_1$ .

La figure 4.7 montrent le résultat de l'application de l'algorithme à l'annulation d'écho acoustique avec l'utilisation de la procédure de détection du full duplex. On note que le signal du deuxième locuteur est introduit après un certain temps pour permettre à l'algorithme de converger et de mettre à jour la constante de seuil. On constate que la détection du full duplex améliore considérablement les performances de l'annuleur d'écho notamment dans les intervalles de temps où les deux signaux sont présents. D'autre part, en comparant ce résultat avec celui de la figure 4.6 on note que pendant les instants où seul le locuteur lointain est présent, les performances des deux structures (avec et sans détection du full duplex) sont identiques, ce qui confirme l'efficacité de la procédure de détection du full duplex.

### 4.3.3 Application dans le cas du full duplex en présence de bruit

Dans cette situation le signal proche est constitué de l'écho, du signal du locuteur lointain et du bruit d'un climatiseur. On discutera dans cette partie l'influence du bruit sur la convergence de l'algorithme.

La figure 4.8 montre les résultats de l'application. On constate que la présence du bruit dégrade les performances de l'annuleur d'écho, néanmoins une réduction notable de l'écho est enregistrée.

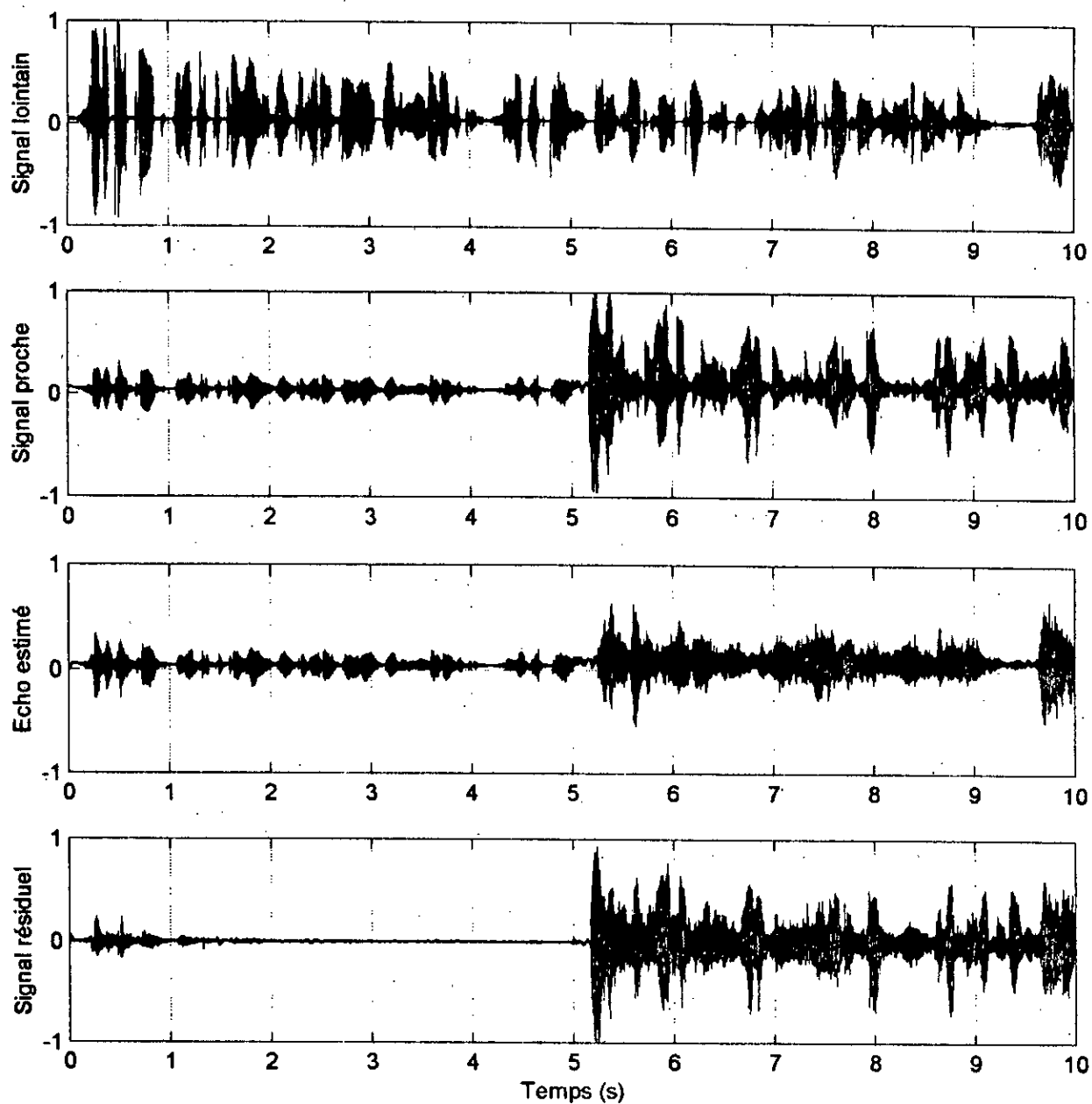
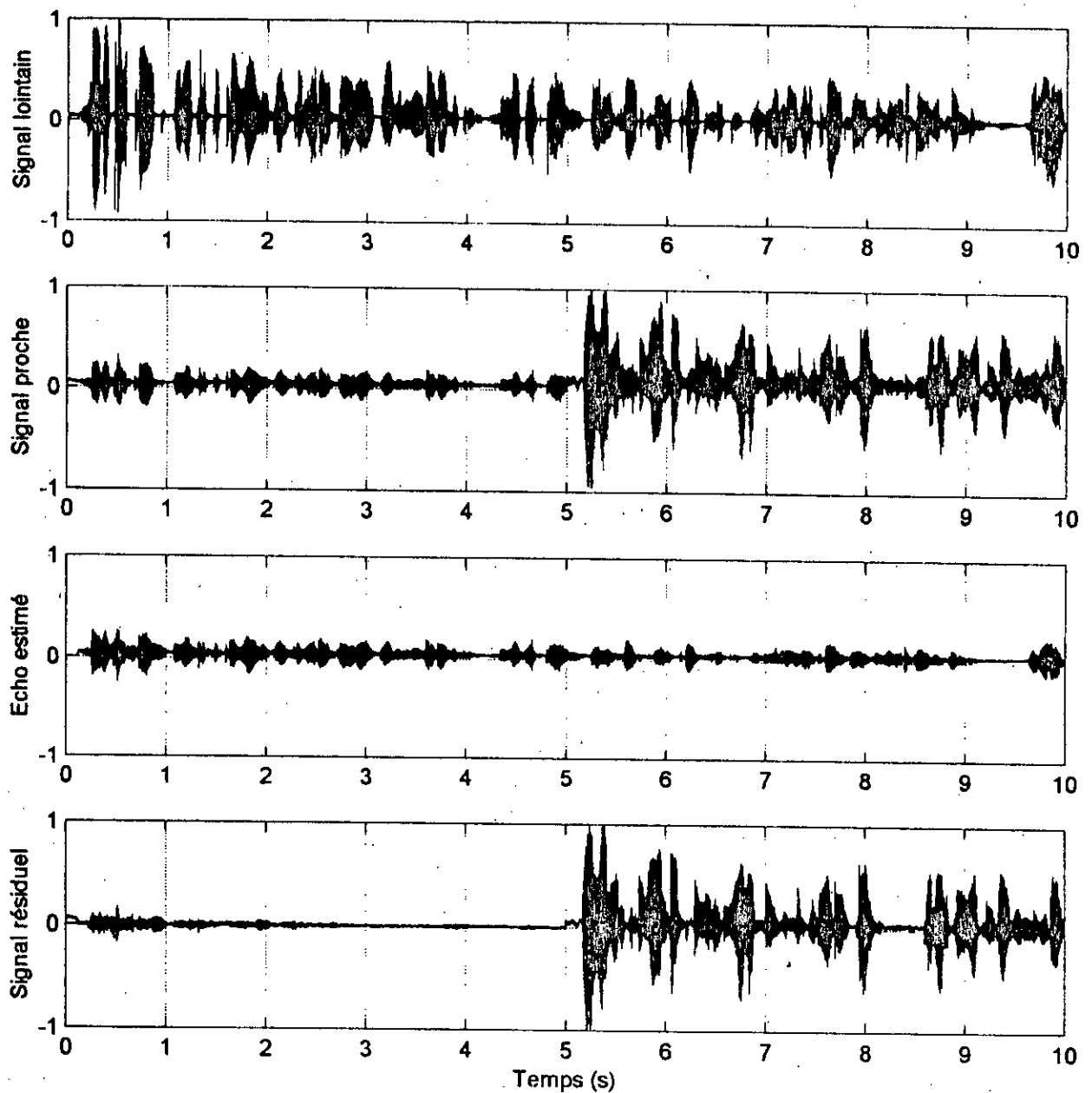


Figure 4.6 : Annulation d'écho acoustique dans le cas "Full duplex"

Algorithme GMDF  $\alpha$  contraint

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92.$$



**Figure 4.7 :** Annulation d'écho acoustique avec détection du full duplex.

*Algorithme GMDF $\alpha$  contraint*

$$L = 1024, K = 4, \alpha = 2, \mu = 0.06, \gamma = 0.92, \lambda_1 = 0.75, \lambda_2 = \lambda_3 = 0.9$$

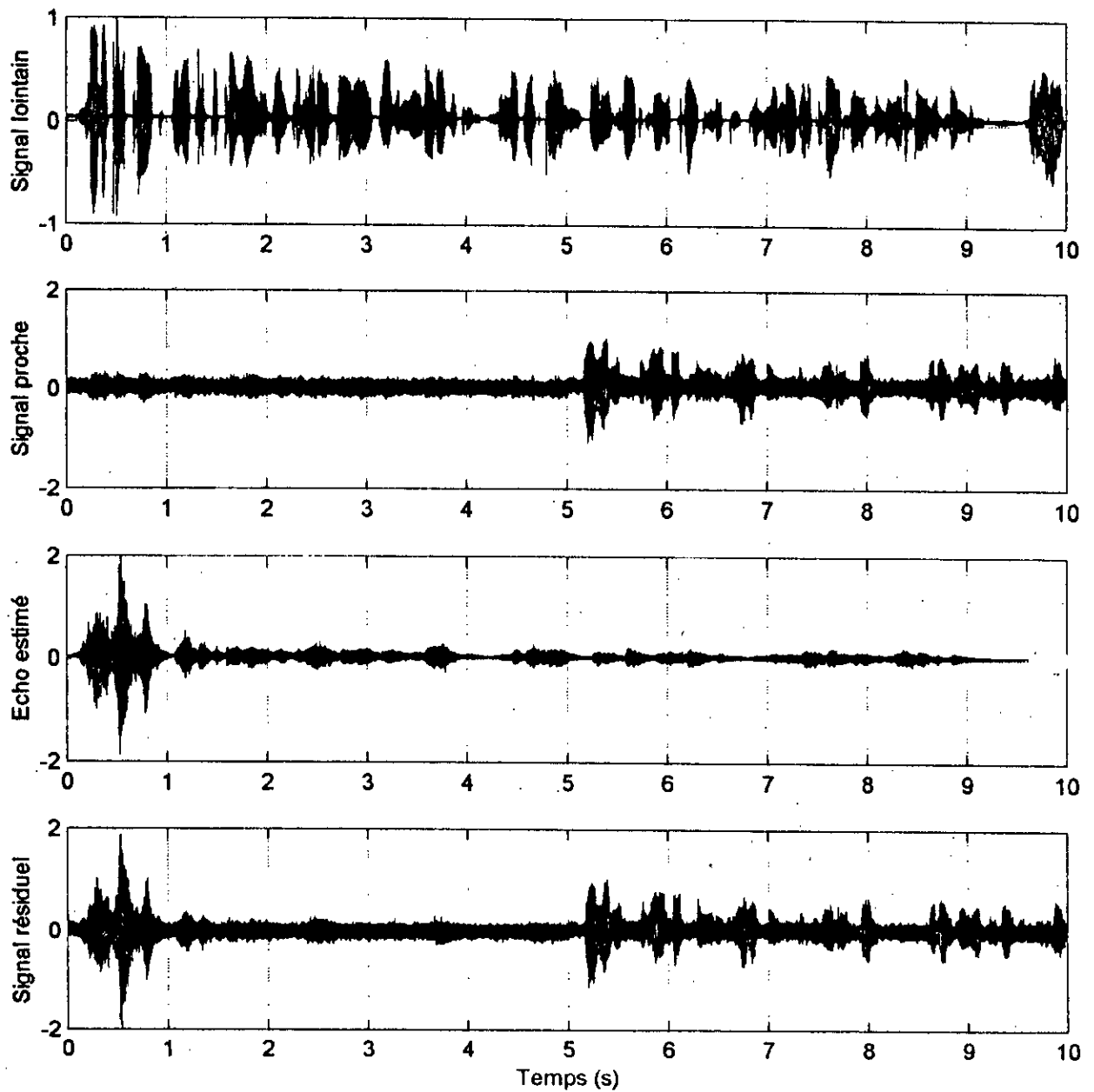


Figure 4.8 : Annulation d'écho acoustique avec détection du full duplex en présence de bruit.

Algorithme GMDF $\alpha$  contraint

$L = 1024$ ,  $K = 4$ ,  $\alpha = 2$ ,  $\mu = 0.06$ ,  $\gamma = 0.92$ ,  $\lambda_1 = 0.75$ ,  $\lambda_2 = \lambda_3 = 0.9$ , RSB = 10 dB

#### 4.4 Conclusion

Les résultats obtenus dans cette étude expérimentale nous ont confirmé l'efficacité de l'algorithme GMDFA en annulation d'écho acoustique. Les expérimentations effectuées ont montré l'importance de la segmentation de la réponse impulsionnelle car elle permet un ajustement du délai de traitement selon les performances souhaitées. D'autre part la détection du full duplex est d'un intérêt certain, car cette procédure permet un fonctionnement autonome du processus adaptatif. L'analyse de la robustesse vis-à-vis du bruit a montré certaines performances qui restent insuffisantes et qui doivent donc être améliorées, ce qui fera l'objet de notre prochain chapitre. Dans les différentes applications réalisées, un écho résiduel est toujours observé. Celui-ci est dû aux distorsions non linéaires introduites par le haut-parleur ainsi qu'à la limitation de la longueur de la réponse impulsionnelle modélisant le canal d'écho. De ce fait il sera très utile d'introduire dans la structure du filtre adaptatif une partie de traitement qui sera destinée à la compensation de ces nonlinéarités.



# Chapitre 5

## Amélioration des performances

Pour les applications de téléphonie mains-libres dans des environnements perturbés par la présence d'un bruit de fond, il est intéressant d'intégrer au système d'annulation d'écho acoustique une procédure de réduction de bruit. Dans ce chapitre, une solution permettant l'amélioration des performances de l'annuleur d'écho acoustique par réduction du bruit accompagnant le signal du locuteur proche sera présentée. De plus, l'écho résiduel pourra être supprimé en considérant ce dernier comme faisant partie du bruit ambiant. La méthode choisie, de part sa simplicité d'implémentation, ainsi que son faible coût en terme de complexité de calcul, permet d'obtenir de bonnes performances.

### 5.1 Introduction

L'amélioration de la qualité des signaux de parole perturbés par un bruit additif est d'un intérêt certain dans plusieurs applications, les plus connues sont :

- Situation de téléphonie mains-libres dans un véhicule, l'utilisateur doit faire face à des niveaux de bruit supérieurs à la normale, et ce par comparaison avec des récepteurs téléphoniques standards, ceci est dû à l'augmentation de la distance séparant l'utilisateur du microphone.
- Utilisation des systèmes contrôlés par la voix comme les radiorécepteurs et les téléphones, ce qui impose l'utilisation d'un système de reconnaissance automatique de la parole, les taux d'identification de ce dernier sont d'autant plus meilleurs que le bruit superposé aux commandes parlées est réduit.

Pour les méthodes à canal unique, qui sont spécialement intéressantes vu la facilité d'installation du microphone, la technique de soustraction spectrale est l'une des plus utilisées. Ceci étant essentiellement dû à sa simplicité d'implémentation et son faible coût en terme de complexité arithmétique, ce qui en fait un choix par excellence dans les applications en temps réel. Cependant, l'inconvénient majeur des méthodes à canal unique réside dans le fait qu'aucune référence du signal de bruit n'est disponible, par conséquent, la DSP (Densité Spectrale de Puissance) du bruit doit être estimée à partir du signal perturbé. Pour ce faire, deux méthodes sont envisageables, la méthode des statistiques minimales [33], et la méthode de détection d'activité vocale [32]. Cette dernière qui se base sur la mise à jour de l'estimée de la puissance du bruit durant l'absence du signal de parole, présente l'inconvénient de nécessiter un processus de détection d'activité vocale, et qui est une tâche difficile à exécuter pour des signaux de parole fortement bruités, de ce fait, cette méthode ne sera pas abordée dans ce qui suit.

## 5.2 Analyse de la suppression soustractive du bruit

A travers cette section, on décrira l'algorithme de soustraction spectrale basée sur l'estimation du bruit par la méthode des statistiques minimales initialement proposée par *R. Martin* [33].

### 5.2.1 Modélisation du bruit additif

Considérons un signal de parole  $p(n)$  dégradé par un bruit additif non corrélé  $b(n)$ , le signal résultant s'exprime par :

$$s(n) = p(n) + b(n) \quad (5.1)$$

Evaluons à présent la matrice d'autocorrélation du signal  $s(n)$  notée  $\Gamma_{ss}(m)$  :

$$\begin{aligned} \Gamma_{ss}(m) &= E\{s(l)s^*(l+m)\} = E\{[p(l) + b(l)][p^*(l+m) + b^*(l+m)]\} \\ &= E\{p(l)p^*(l+m)\} + E\{b(l)b^*(l+m)\} + E\{p(l)b^*(l+m)\} + E\{b(l)p^*(l+m)\} \end{aligned} \quad (5.2)$$

Avec l'hypothèse que le bruit  $b(n)$  est non corrélé avec le signal de parole  $p(n)$ , les deux termes  $E\{p(l)b^*(l+m)\}$  et  $E\{b(l)p^*(l+m)\}$  sont nuls, et on aura :

$$\Gamma_{ss}(m) = \Gamma_{pp}(m) + \Gamma_{bb}(m) \quad (5.3)$$

En prenant les transformées de Fourier de part et d'autre de l'équation (5.3), on aboutit à [34] :

$$S_{DSP}(k, sn) = P_{DSP}(k, sn) + B_{DSP}(k, sn) \quad (5.4)$$

où  $S_{DSP}(k, sn)$ ,  $P_{DSP}(k, sn)$  et  $B_{DSP}(k, sn)$  sont les densités spectrales de puissance à court terme des signaux d'entrée, de parole et de bruit respectivement, à la fréquence  $k$  et à l'instant  $sn$ ,  $s$  étant l'indice de la fenêtre considérée.

### 5.2.2 Estimateur de soustraction spectrale

Le filtre de soustraction spectrale qu'on notera  $G(k, sn)$  est déterminé en estimant la DSP du signal du bruit dont la procédure de détermination sera décrite plus loin. En désignant la DSP estimée du signal de parole par  $\hat{P}_{DSP}(k, sn)$  et celle du bruit par  $\hat{B}_{DSP}(k, sn)$ , on a :

$$\hat{P}_{DSP}(k, sn) = S_{DSP}(k, sn) - \hat{B}_{DSP}(k, sn) \quad (5.5)$$

ou bien :

$$\hat{P}_{DSP}(k, sn) = \left( 1 - \frac{\hat{B}_{DSP}(k, sn)}{S_{DSP}(k, sn)} \right) S_{DSP}(k, sn)$$

d'où l'on déduit que :

$$\hat{P}_{DSP}(k, sn) = G(k, sn) S_{DSP}(k, sn)$$

avec :

$$G(k, sn) = 1 - \frac{\hat{B}_{DSP}(k, sn)}{S_{DSP}(k, sn)} \quad (5.6)$$

### 5.2.3 Rectification demi-onde

Il est important de noter que l'équation (5.5) ne garantit en aucune manière que le terme  $\hat{P}_{DSP}(k, sn)$  soit positif, par conséquent, les valeurs négatives doivent être annulées. Cette modification peut être simplement implémentée par une rectification, dite demi-onde, de  $G(k, sn)$ . L'estimateur devient alors [32] :

$$G_{Boll}(k, sn) = \max \left( 1 - \frac{\hat{B}_{DSP}(k, sn)}{S_{DSP}(k, sn)}, 0 \right), \quad k = 0, \dots, L-1 \quad (5.7)$$

où  $L$  est la taille de la FFT.

L'inconvénient de cette procédure se manifeste essentiellement par l'élimination éventuelle de certaines composantes spectrales du signal de parole pour des RSB faibles, ce qui conduit à une diminution de l'intelligibilité du signal. Une amélioration de la fonction de transfert exprimée par l'équation (5.7) sera décrite plus loin.

### 5.2.4 Estimation de la DSP du bruit

L'estimation de la DSP du bruit peut être obtenue en détectant, à chaque composante spectrale, les valeurs minimales de la DSP à court terme du signal bruité. Pour un signal de parole non bruité, la DSP présenterait des amplitudes nulles à certaines composantes spectrales ou pendant des instants de silence donnés. Par conséquent, la présence du bruit à chaque mode unique, se manifeste par une certaine excentricité plus ou moins permanente en fonction de la stationnarité du bruit additif, ceci en se basant sur l'hypothèse que le bruit de fond est stationnaire sur de courts intervalles temporels. Autrement dit, l'estimation de la DSP du bruit s'effectue en prenant le minimum de chaque composante de la DSP à court terme du signal bruité (mode par mode) sur un intervalle de temps discret de durée donnée appelé taille du filtre de minimisation et noté  $\tau$ . Ainsi cette méthode se résume comme suit :

- Estimation de la DSP du signal bruité en utilisant un periodogramme lissé exponentiellement de manière récursive avec une constante de lissage  $\alpha$  [27] :

$$S_{DSP}(k, sn) = \alpha S_{DSP}(k, sn-1) + (1-\alpha)|S(k, sn)|^2 \quad (5.8)$$

où la constante de lissage  $\alpha$  est comprise dans l'intervalle  $[0.3, 0.7]$ .

- Estimation de la DSP du bruit par détection des minimums de la puissance du signal d'entrée sur des fenêtres de  $M$  buffers de minimisation :

$$\left. \begin{aligned} BM_n(k, j) &= S_{DSP}(k, sn) & \text{si } j &= 1 \\ \text{et :} & & & \\ BM_n(k, j) &= BM_{n-1}(k, j-1) & \text{si } j &= 2, \dots, M \\ BM_n(k, 1) &= \min(BM_{n-1}(k, 1), S_{DSP}(k, sn)) & & \text{: sinon} \end{aligned} \right\} \text{ pour : } n \equiv 0 \pmod{\tau}$$

Et finalement :

$$\hat{B}_{DSP}(k, sn) = \min(BM_n(k, 1), BM_n(k, 2), \dots, BM_n(k, M))$$

où  $BM_n(k, j)$  est une matrice  $L \times M$  dont les colonnes représentent les  $M$  buffers de minimisation et  $k$  l'indice fréquentiel avec  $k = 0, \dots, L-1$ .

Un bon choix du nombre de buffers de minimisation à utiliser est un compromis entre une grande mémoire utile pour la mémorisation des données correspondantes et une modulation du signal de sortie par le bruit estimé.

### 5.2.5 Fonction de pondération modifiée

Une méthode plus efficace pour la suppression de bruit et des tonalités musicales résiduelles consiste en l'application de la fonction de pondération modifiée donnée par [35 - 36] :

$$G_{\text{mod}}(k, sn) = \max \left( 1 - \sqrt{\beta \frac{\hat{B}_{DSP}(k, sn)}{S_{DSP}(k, sn)}}, ss \right) \quad (5.9)$$

Comparée avec l'équation (5.7), on remarque deux paramètres supplémentaires, le facteur de surestimation  $\beta$ , et le seuil spectral minimal  $ss$ . Le facteur de surestimation augmente le niveau du bruit estimé dans le but de prévenir les fluctuations à court terme de la DSP du bruit, permettant ainsi de réduire significativement l'effet des tonalités musicales. En pratique, un bon choix du facteur de surestimation se situe dans l'intervalle  $[0.9, 4]$  [35]. Cependant, ce choix influe directement sur la qualité du signal de sortie. Ainsi de grandes valeurs conduisent à une bonne élimination du bruit, évitant de la sorte la manifestation des tonalités musicales dues aux variations rapides du signal de bruit avant soustraction spectrale, en revanche, ceci a pour conséquence une élimination des composantes spectrales de faible énergie du signal de parole et par suite une dégradation de l'intelligibilité. Ainsi, l'introduction du seuil spectral minimal se justifie par le fait que ce dernier permet de préserver la sonorité naturelle du signal de parole, et qui de plus permet de masquer les tonalités musicales restantes avec le bruit résiduel durant l'absence d'activité vocale. Une valeur typique de ce paramètre se situe entre 0.02 et 0.1 [35]. L'inconvénient majeur dans ce cas se manifeste par une limitation de l'amélioration du RSB à  $-20 \log(ss)$  dB. Une solution à ce problème consiste à utiliser le seuil modifié :

$$ssm(k, sn) = ss \sqrt{\frac{\hat{B}_{DSP}(k, sn)}{|S(k, sn)|^2}} \quad (5.10)$$

Dans ce cas, l'amélioration du RSB devient :

$$\Delta RSB = -20 \log(ss) + RSB_{a \text{ priori}}$$

avec :

$$RSB_{a\ priori} = 10 \log \left( \frac{|S(k, sn)|^2}{\hat{B}_{DSP}(k, sn)} \right)$$

Ainsi, le seuil spectral minimal sera faible pour un  $RSB_{a\ priori}$  élevé et vice versa.

### 5.3 Implémentation de l'algorithme

En se basant sur les développements de la section précédente, un algorithme complet d'analyse et de synthèse des données peut être construit. On présente à travers cette section les spécifications requises pour l'implémentation du système de soustraction spectrale pour la suppression de bruit.

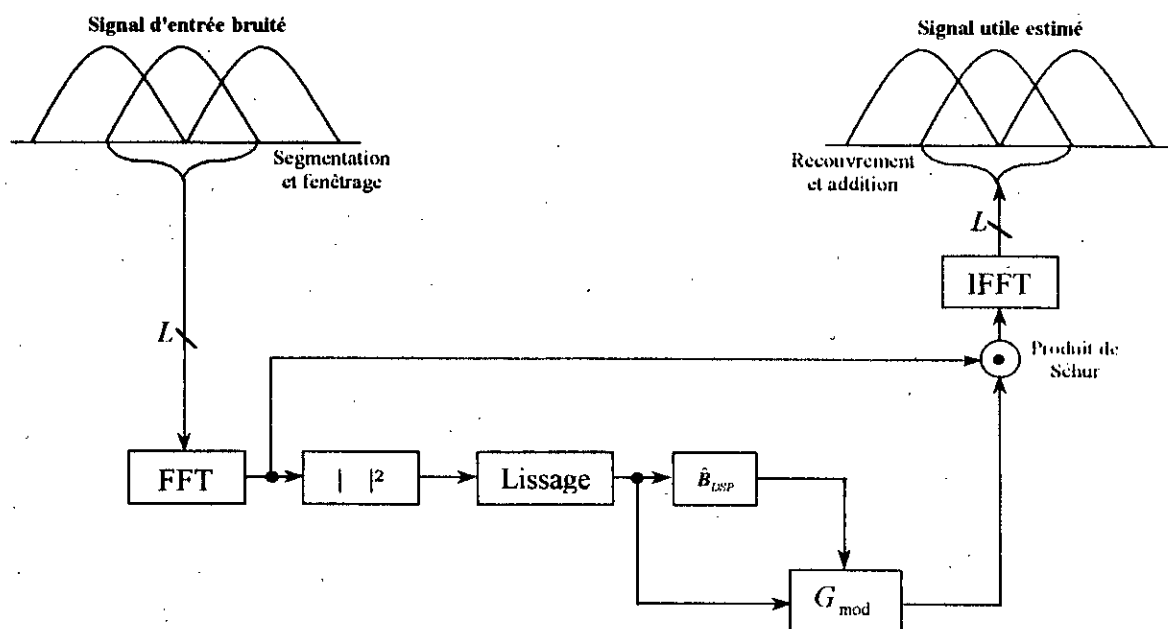


Figure 5.1 : Diagramme bloc du système réducteur de bruit

Le signal de parole bruité est segmenté et fenêtré de manière à ce qu'en absence de modifications spectrales (dues à l'opération de filtrage), la synthèse du signal de parole par recouvrement-addition des segments successifs aboutit à une fonction de transfert globale du système égale à l'identité. Ainsi, les données sont segmentées et fenêtrées en exploitant le fait qu'une séquence décomposée en segments obtenus par recouvrement à 50 %, et par multiplication de chaque segment par une fenêtre de Hanning, la somme de ces séquences fenêtrées aboutit à la séquence originale. L'hypothèse de stationnarité du signal de parole sur de courts intervalles temporels nous conduit au choix de la durée de chaque segment prise égale à

32 ms [32 et 35]. Ainsi le passage d'un état stationnaire à un autre est réalisé à chaque itération. Les tests réalisés lors des différentes simulations confirment par ailleurs la validité de ce choix. Ainsi, pour une fréquence d'échantillonnage de 8 kHz, une fenêtre de  $L=256$  points décalée à chaque étape de 128 points est utilisée. La figure 5.1 illustre la procédure de segmentation et de fenêtrage des données.

Une FFT sur  $L$  points de chaque fenêtre est réalisée, ensuite le carré du module de chaque composante obtenue est calculé. Un lissage exponentiel est par la suite effectué sur chaque amplitude en utilisant l'équation 5.8. L'estimée de la DSP du bruit est déterminée en utilisant l'estimateur des statistiques minimales. La fonction de pondération modifiée est par la suite calculée et finalement le signal de sortie sera synthétisé par recouvrement-addition des segments adjacents (voir figure 5.1) :

$$\hat{P}(k, sn) = G_{\text{mod}}(k, sn)S(k, sn) \quad ; \text{ pour } : k = 0, \dots, L-1 \quad (5.11)$$

où  $S(k, sn)$  est le  $k^{\text{ème}}$  élément de  $S_{sn}$  qui est la TFD du segment d'entrée fenêtré à l'instant  $sn$  :

$$S_{sn} = W_L s_{sn}$$

avec :

$$s_{sn} = f \odot [s(sn-L+1), \dots, s(sn)]^T$$

où :  $f$  est une fenêtre de Hanning de taille  $L$ .

Finalement, le segment correspondant du signal de sortie est obtenu par l'application d'une TFD inverse :

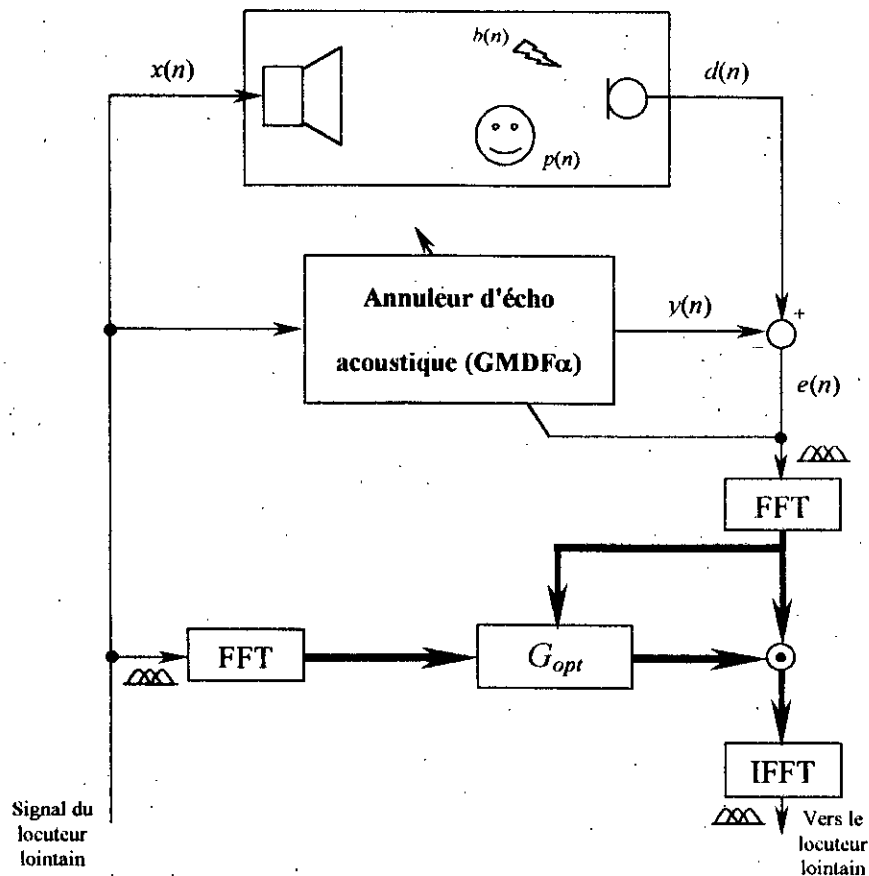
$$\hat{p}_{sn} = W_L^{-1} \hat{P}_{sn} \quad (5.12)$$

où :  $\hat{P}_{sn} = [\hat{P}(0, sn), \dots, \hat{P}(L-1, sn)]^T$

#### 5.4 Atténuation de l'écho résiduel

Quoique l'annulation d'écho acoustique par utilisation de l'algorithme GMDF $\alpha$  donne de bons résultats dans des environnements non perturbés par un bruit ambiant, les performances obtenues en présence de bruit se trouvent nettement dégradées. Afin d'améliorer la robustesse du système et ce dans le but d'atteindre au moins l'atténuation de l'écho minimale requise par les organismes internationaux de normalisation (par exemple l'International Telecommunications Union [8]), nous introduisons à travers cette section une procédure de réduction de bruit et de suppression de l'écho résiduel. Les performances ainsi obtenues sont acceptables même dans une situation de full duplex.

Usuellement, on admet que le bruit ambiant est stationnaire au sens large, ce qui conduit à une estimée à variation lente de la DSP. En ce qui concerne l'écho résiduel, il n'est plus possible d'utiliser l'estimateur des statistiques minimales. Ceci se justifie par la non stationnarité typique des signaux de parole. De plus, assurer une bonne distinction entre le signal utile qu'est le signal de parole du locuteur proche et le signal d'écho résiduel ne peut être réalisée par l'utilisation de cette méthode.



**Figure 5.2 :** Application de la soustraction spectrale à la réduction de bruit et la suppression de l'écho résiduel

Dans la mesure où l'on dispose d'une information non perturbée du signal du locuteur lointain, on considère la DSP de ce dernier comme une approximation des caractéristiques spectrales de l'écho résiduel à un facteur multiplicatif près (en absence d'activité vocale du locuteur proche). Le facteur d'atténuation en puissance  $\gamma$  peut être exprimé par :

$$\gamma = \frac{E\{(d(n) - y(n))^2\}}{E\{(x(n))^2\}} \quad (5.11)$$



où  $y(n)$  représente le signal de l'écho estimé,  $x(n)$  le signal du locuteur lointain et  $d(n)$  le signal de l'écho réel non perturbé par le signal du locuteur proche, ainsi une bonne estimation du facteur  $\gamma$  doit être effectuée pendant un certain temps de silence du locuteur proche, ce qui peut être réalisé dès la mise en marche du système pour une initialisation préliminaire, par la suite la mise à jour de ce paramètre peut être effectuée en absence d'activité vocale du locuteur proche.

L'estimation de la DSP à court terme du signal lointain  $X_{DSP}(k, sn)$  est réalisée de façon analogue à l'équation (5.8) avec la même constante de lissage :

$$X_{DSP}(k, sn) = \alpha X_{DSP}(k, sn-1) + (1-\alpha)|X(k, sn)|^2 \quad (5.12)$$

où  $X(k, sn)$  représente le coefficient de la TFD à court terme du signal  $x(n)$  respectivement à la fréquence  $k$  et à l'instant  $sn$ . On doit noter par ailleurs que l'estimateur de l'équation (5.12) ne prend pas en considération les distorsions du signal  $x(n)$  dues au chemin de l'écho acoustique, cependant au vu des résultats obtenus, on a pu déduire qu'on obtient une estimation acceptable de l'écho résiduel.

En considérant l'écho résiduel comme faisant partie du signal de bruit, nous utilisons la fonction de transfert optimale suivante [36] :

$$G_{opt}(k, sn) = \max \left( 1 - \sqrt{\frac{\beta \hat{B}_{DSP}(k, sn) + \gamma X_{DSP}(k, sn)}{S_{DSP}(k, sn)}}, ssm(k, n) \right) \quad (5.13)$$

## 5.5 Résultats expérimentaux

Afin de mettre en évidence l'efficacité de l'algorithme, tant du point de vue réduction de bruit et suppression de l'écho résiduel que préservation de la qualité sonore du signal de parole à filtrer, nous avons prélevé plusieurs séquences de parole enregistrées dans des environnements caractérisés par différents types de bruits. On décrit à travers cette section les résultats des différents tests réalisés dans des situations de simple activité vocale (half duplex) et dans une situation de double activité vocale (full duplex).

### 5.5.1 Réduction de bruit dans le cas du simple duplex

La première expérience consiste à effectuer un enregistrement d'une séquence de parole dans un environnement non bruité échantillonné à la fréquence de 11 kHz, puis on y ajoute un

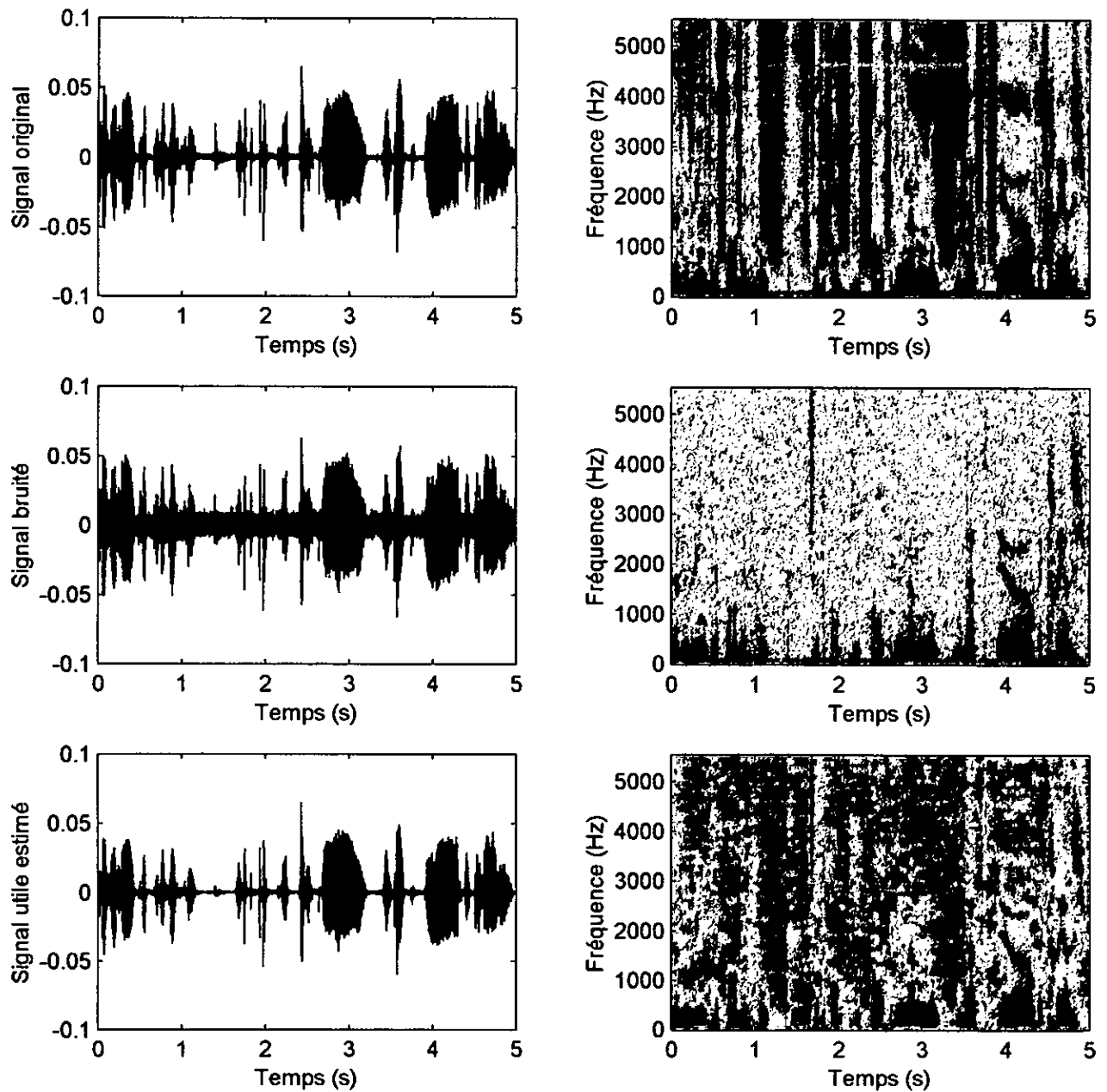


Figure 5.3 : Réduction de bruit blanc gaussien de moyenne nulle et de variance

$$\sigma^2=0.003 \text{ (RSB}=12 \text{ dB)}$$

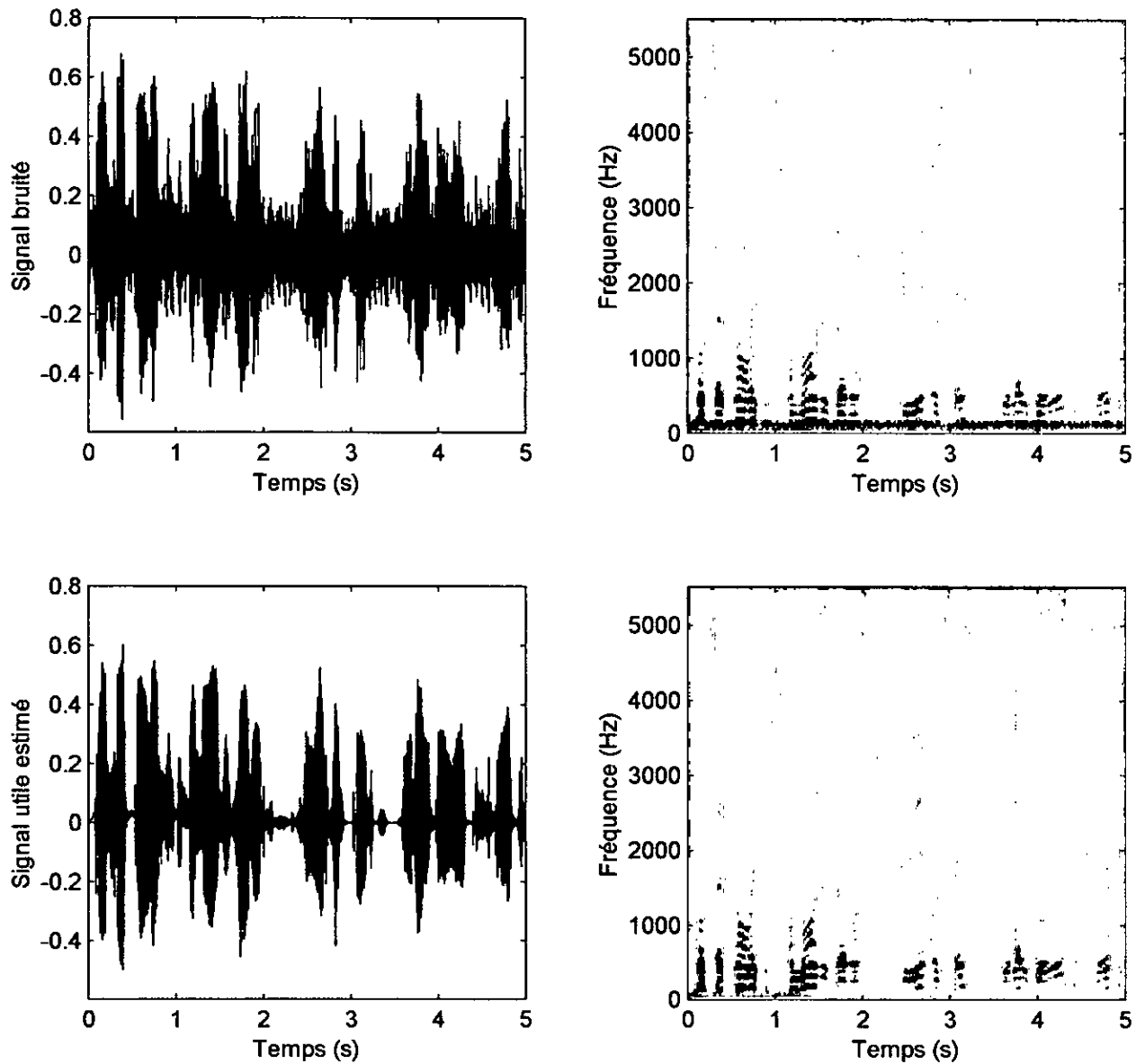


Figure 5.4 : Réduction de bruit dans une automobile (RSB=10 dB)

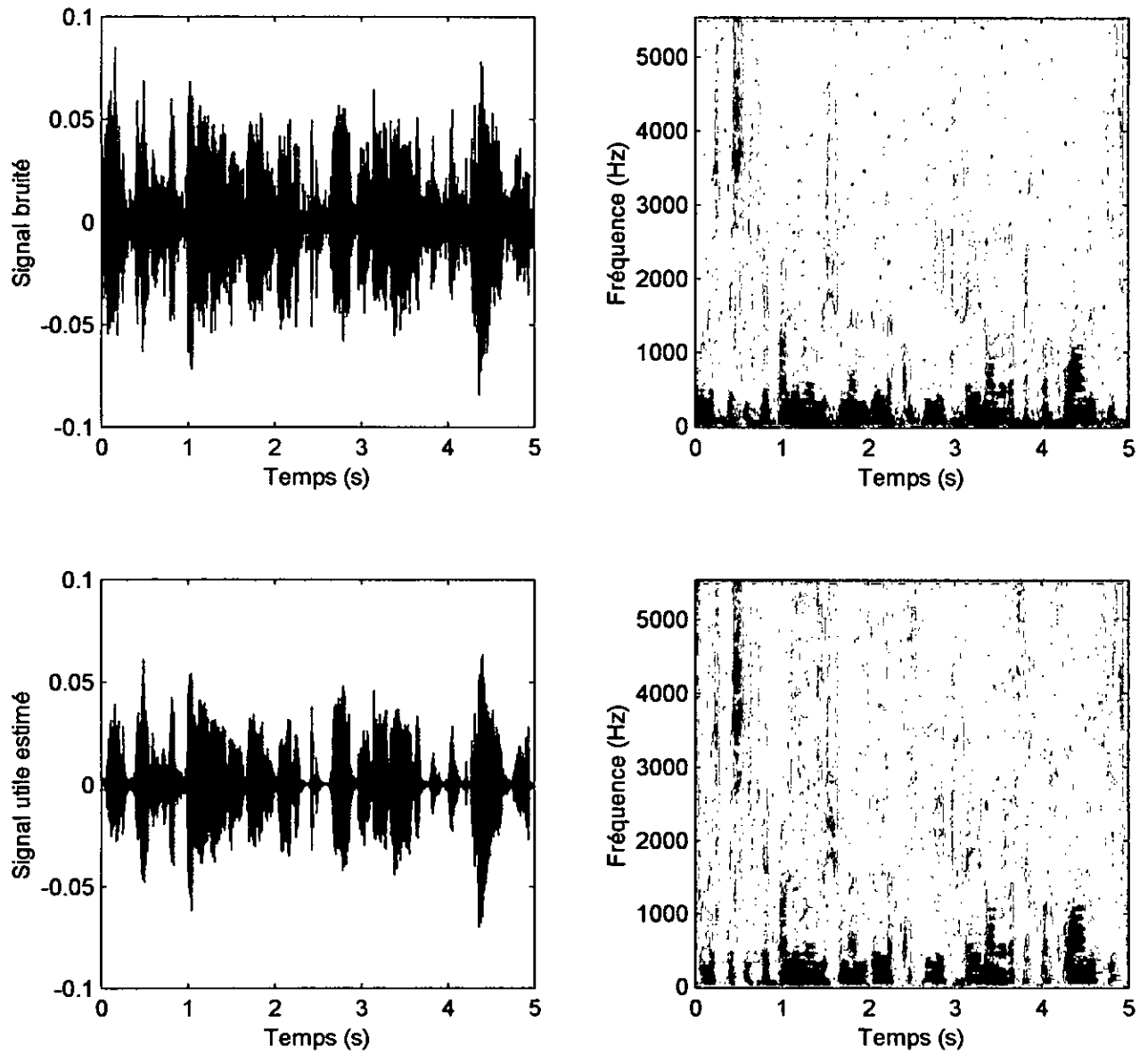


Figure 5.5 : Réduction de bruit de climatisation dans un bureau ( $RSB=20\text{ dB}$ )

bruit blanc gaussien de moyenne nulle et de variance 0.003. On obtient de la sorte un RSB égal à 12 dB. Le signal obtenu considéré comme signal de parole du locuteur proche, est par la suite filtré par l'algorithme réducteur de bruit sans pour autant être affecté par l'annuleur d'écho, ni par l'estimateur d'écho résiduel, ceci en absence du signal du locuteur lointain. On représente à la figure 5.3, respectivement de haut en bas, le signal original, le signal bruité obtenu ainsi que le signal de parole filtré après convergence, et de gauche à droite les représentations temporelles ainsi que les spectrogrammes. On remarque dans ce cas, que le signal obtenu est relativement atténué et présente certaines amplitudes spectrales uniformément distribuées sur le plan temps-fréquence (points jaunes). A l'écoute du signal résultant celles-ci se manifestent par des tonalités musicales plus ou moins gênantes. En revanche, on note une bonne intonation du signal de parole avec une intelligibilité parfaitement conservée.

Une deuxième expérience similaire consiste à enregistrer une séquence de parole dans un environnement perturbé par un bruit de moteur d'automobile, avec un RSB de 10 dB. Les signaux obtenus sont représentés en figure 5.4. On note sur le signal bruité la présence des composantes spectrales du bruit caractérisées cette fois par des modes bien distingués aux fréquences 800 Hz et 1.6 kHz. D'autre part, on remarque sur le spectrogramme du signal filtré la disparition de la composante continue après une durée d'adaptation de 1.5 s environs. Dans cette situation, la performance de l'algorithme est bien mise en évidence par une bonne suppression du bruit d'une part, et une distorsion du signal utile quasi imperceptible d'autre part.

Une troisième expérience consiste à enregistrer un signal de parole perturbé par la présence d'un bruit ambiant dû au ventilateur d'un climatiseur de bureau. La manifestation de cette perturbation est accentuée par un rapprochement exagéré du microphone près de la source de bruit. Le signal bruité ainsi que le signal filtré obtenus sont représentés à la figure 5.5. On note une bonne suppression du bruit. A l'écoute, on perçoit une conservation de la sonorité naturelle du signal de parole.

### 5.5.2 Suppression d'écho résiduel dans le cas du simple duplex

Toujours dans une situation de simple duplex, on envisage cette fois de tester les performances de l'estimateur de l'écho résiduel. Dans ce cas, l'influence du système réducteur de bruit est inhibée par le choix d'un facteur de surestimation de la DSP du bruit strictement nul. On présente par la figure 5.6 les résultats obtenus avec et sans suppression d'écho résiduel. Ces résultats sont obtenus pour une séquence de parole d'une durée de 10 s échantillonnée à une

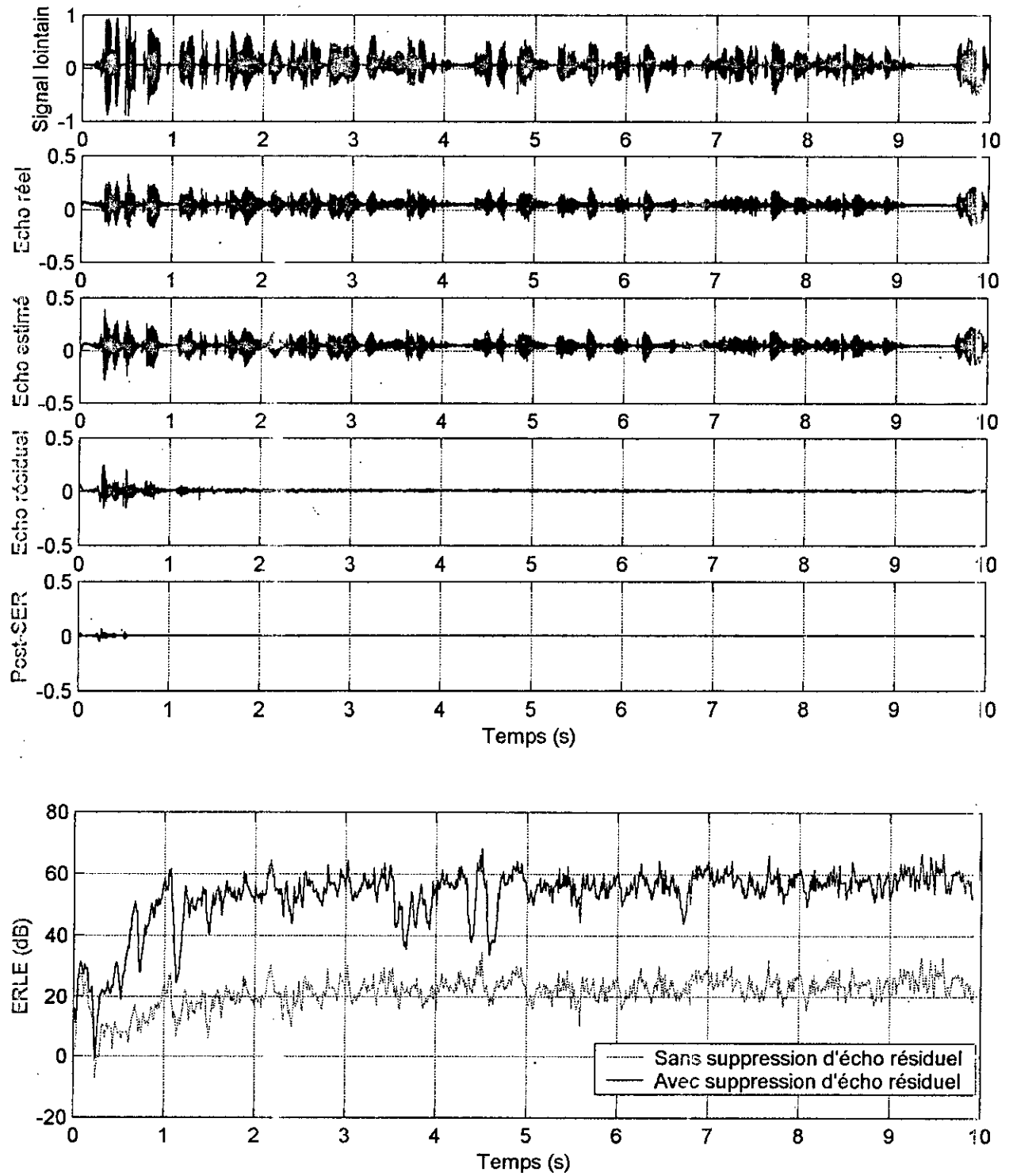


Figure 5.6 : Situation de simple duplex avec suppression d'écho résiduel ( $RSB = 30 \text{ dB}$ )

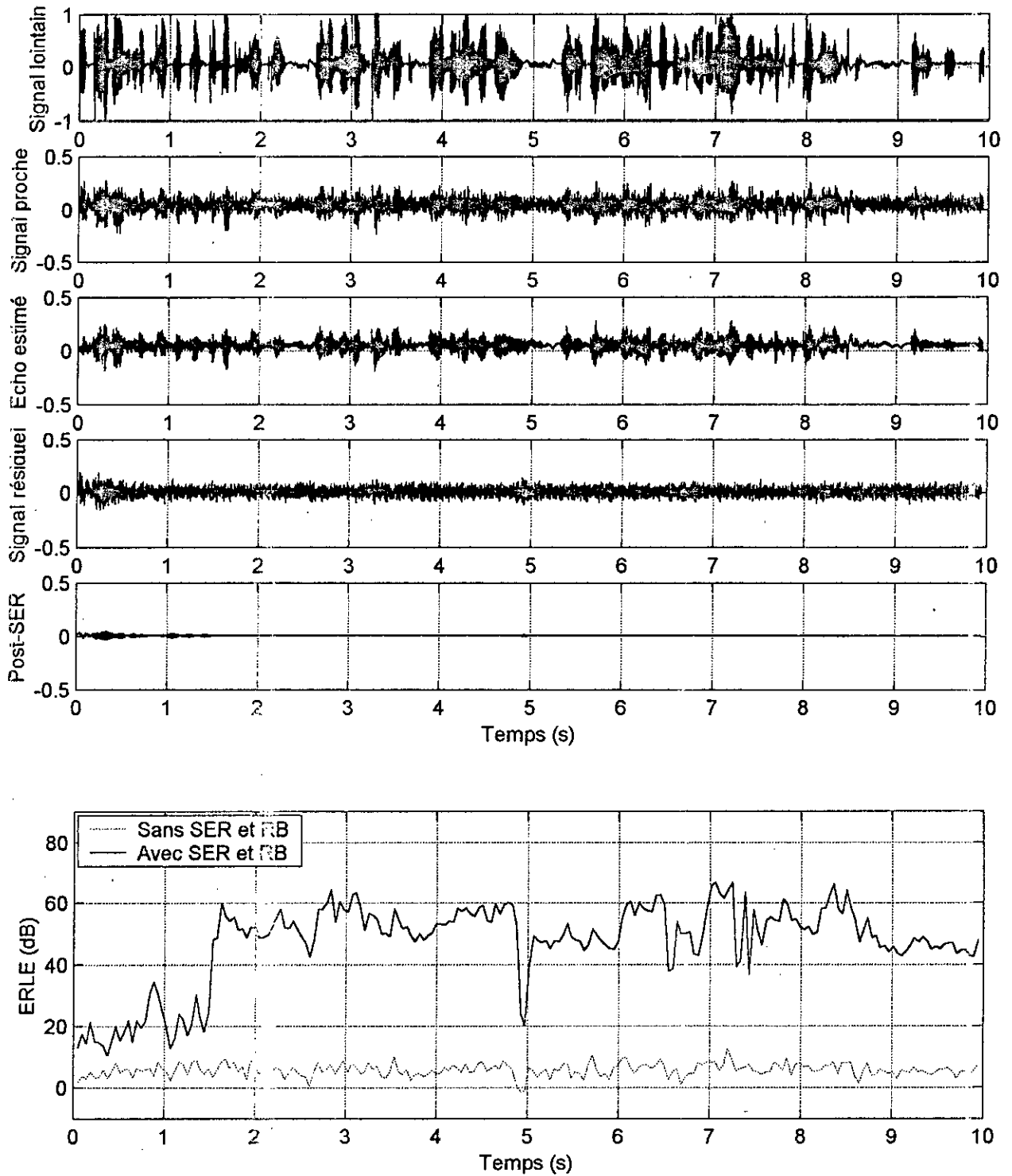


Figure 5.7 : Situation de simple duplex avec reduction du bruit d'automobile et suppression d'écho résiduel (RSB = 10 dB)

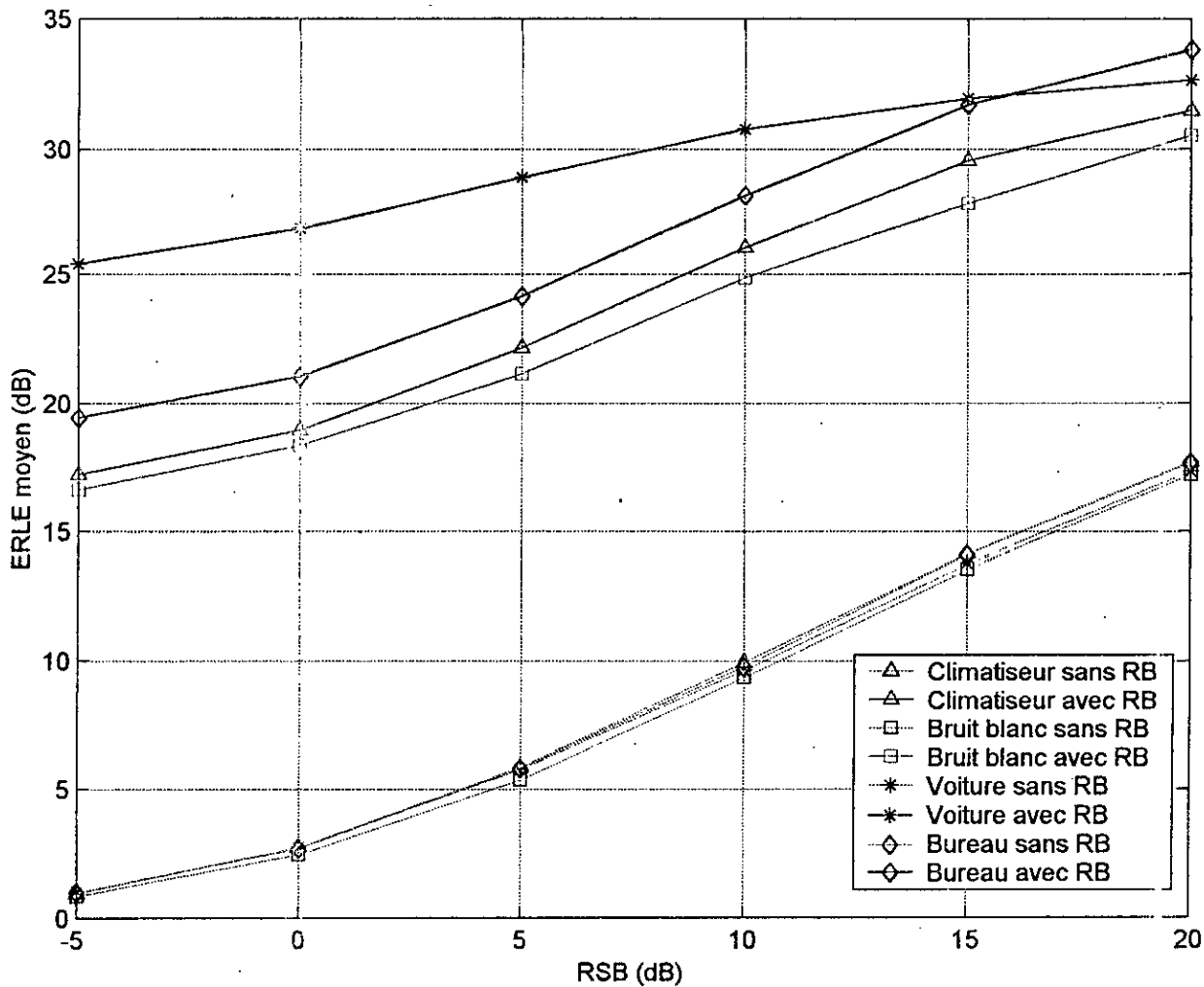


Figure 5.8 : *ERLE* moyen avant et après réduction de bruit et suppression d'écho résiduel dans le cas de simple duplex



fréquence de 11 kHz, une réponse impulsionnelle tronquée à 1024 coefficients et segmentée avec un facteur  $K=4$ , un facteur de suréchantillonnage  $\alpha=2$ , un pas d'adaptation égal à 0.04 et enfin un facteur de lissage de 0.92. On note une annulation quasi totale du signal d'écho avec une augmentation du ERLE en moyenne de 31 dB, aboutissant ainsi à un ERLE moyen après SER (Suppression d'Echo Résiduel) de 52 dB. On obtient ainsi un résultat en conformité avec les recommandations de l'ITU dans une situation de simple duplex (minimum de 45 dB [8]).

### 5.5.3 Réduction de bruit et suppression d'écho résiduel dans le cas du simple duplex

L'étape suivante consiste à tester la performance de l'algorithme dans une situation de suppression de l'écho résiduel et de réduction du bruit de fond. L'expérience est effectuée avec les mêmes paramètres que précédemment en ajoutant un bruit de fond dû au moteur d'une voiture de tourisme, nous obtenons ainsi un RSB de 10 dB. Toujours en absence d'activité vocale du locuteur proche, nous constatons d'après la figure 5.7 une bonne suppression du signal résiduel avec une augmentation du ERLE de 20 dB en moyenne, aboutissant ainsi à une moyenne finale après convergence de 40 dB. Ce résultat peut être considéré comme acceptable dans les conditions de bruit imposées.

On présente la figure 5.9 résumant les résultats obtenus pour quatre types de bruits ambiants et pour différents RSB de -5 à 20 dB, à chaque valeur de ce dernier on présente la valeur du ERLE correspondante avant et après réduction de bruit avec suppression d'écho résiduel. On constate une amélioration d'au moins 15 dB pour de faibles RSB et de 13 dB pour les grandes valeurs de ce dernier.

### 5.5.4 Réduction de bruit et suppression de l'écho résiduel dans le cas du full duplex

On teste par la suite l'algorithme dans un le cas d'une communication en full duplex avec détection de double activité vocale, réduction de bruit et suppression d'écho résiduel. L'introduction du signal du deuxième locuteur est effectuée après une durée d'environ 5 s, et ce par souci d'assurer une convergence optimale du  $GMD\alpha$ . Les signaux obtenus sont représentés en figure 5.8. On constate un signal perturbateur résiduel minime par rapport au signal du locuteur proche après convergence de 1.5 s imposée essentiellement par la taille du filtre de minimisation. Par ailleurs, on constate une bonne qualité sonore du signal transmit avec une disparition presque totale de l'effet des tonalités musicales, ce dernier étant indiscernable durant l'activité vocale du locuteur proche.

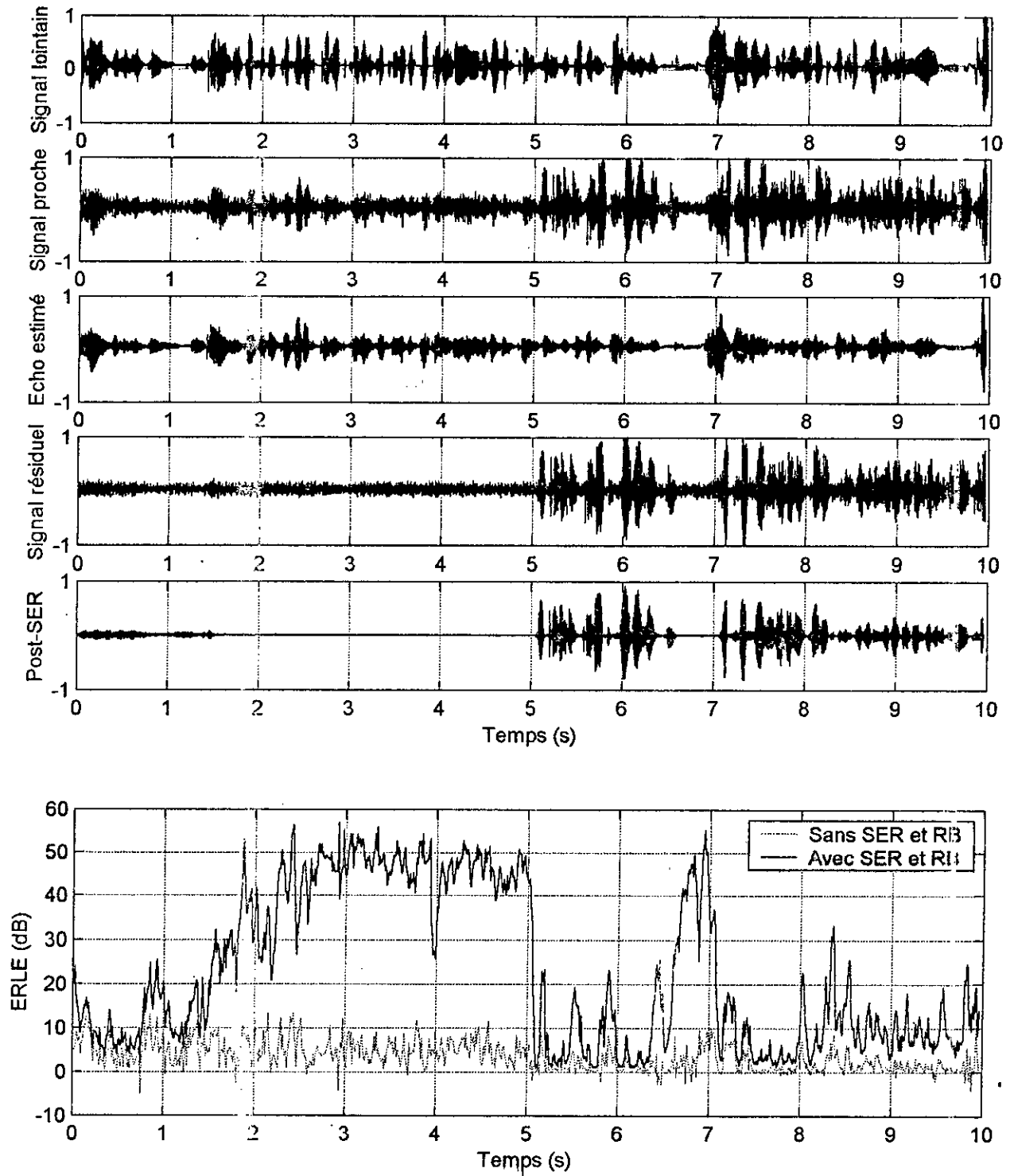


Figure 5.9 : Situation de full duplex avec détection de double activité vocale,  
réduction de bruit et suppression d'écho résiduel  
(RSB = 10 dB)

## 5.6 Conclusion

Nous avons présenté dans ce bref chapitre une amélioration des performances de l'annuleur d'écho acoustique, par la mise à profit de la technique de soustraction spectrale pour la réduction du bruit ambiant et la suppression de l'écho résiduel. Une telle approche est incontournable dans la mesure où les performances des annuleurs d'écho sont fortement affectées par la présence de bruit. Les résultats expérimentaux obtenus affirment par ailleurs la validité de cette approche dans les situations réelles, permettant ainsi d'aboutir à des performances nettement meilleures et qui de plus sont en conformité avec les exigences du marché des télécommunications.

## Conclusion générale

Nous avons présenté par le présent travail un dispositif de protection contre l'écho acoustique et le bruit ambiant dans un contexte de téléphonie mains-libres. L'étude théorique s'est basée sur la modélisation linéaire du chemin d'écho ainsi que sur les méthodes d'optimisation adaptative du modèle par l'utilisation des algorithmes de filtrage adaptatif dans le domaine fréquentiel.

Les résultats expérimentaux obtenus ont montré l'efficacité de l'algorithme choisi pour une telle application. Une amélioration du dispositif a été apportée d'une part par l'introduction d'une procédure de détection de double activité vocale assurant ainsi un fonctionnement autonome du dispositif en full duplex, et d'autre part par l'adjonction d'une structure de réduction de bruit et de suppression de l'écho résiduel. Les résultats satisfaisants ainsi obtenus sont encourageants quant à une future implémentation en temps réel. Néanmoins un perfectionnement pourra être apporté par l'introduction d'une procédure de compensation des non linéarités liées au chemin électrique du signal d'écho.

## Annexe A

### Condition de convergence de l'algorithme LMS

Lors de la conception d'un filtre adaptatif à réponse impulsionnelle finie, le but est de trouver le vecteur  $\mathbf{h}_n$  à l'instant  $n$  minimisant la fonction quadratique :

$$\xi(n) = E \{ |e(n)|^2 \} \quad (\text{A.1})$$

Bien que le vecteur qui minimise  $\xi(n)$  peut être trouvé en posant que la dérivée de  $\xi(n)$  par rapport à  $h^*(k)$  est égale à zéro, une autre approche consiste à rechercher la solution par la méthode du *gradient stochastique*, utilisée pour la recherche des extrêmes de fonctions non-linéaires [5]. L'idée de base de cette méthode est résumée dans ce qui suit.

Soit  $\mathbf{h}_n$  une estimation du vecteur qui minimise l'erreur quadratique moyenne  $\xi(n)$  à l'instant  $n$ . A l'instant  $n+1$  une nouvelle estimation est formée en ajoutant une correction à  $\mathbf{h}_n$  de façon à le faire tendre vers la solution optimale. La correction nous mène à prendre un pas  $\mu$  dans la direction du maximum descendant (*steepest descent*) à travers la surface d'erreur quadratique. Cette direction est donnée par le gradient, qui est un vecteur dont les composantes sont données par les dérivées partielles de  $\xi(n)$  par rapport aux coefficients  $h(k)$ .

$$\nabla \xi(n) = \begin{bmatrix} \partial \xi(n) / \partial h(0) \\ \partial \xi(n) / \partial h(1) \\ \vdots \\ \partial \xi(n) / \partial h(L-1) \end{bmatrix} \quad (\text{A.2})$$

Pour tout vecteur  $\mathbf{h}$ , le gradient est orthogonal à la tangente des lignes pour lesquelles  $\xi(n)$  est constante. Cependant, dans la mesure où le vecteur gradient  $\nabla \xi(n)$  se dirige dans le sens

du maximum ascendant, la direction du maximum descendant est l'inverse de celle du gradient.

Dès lors, l'équation de mise à jour du vecteur  $\mathbf{h}_n$  sera :

$$\mathbf{h}_{n+1} = \mathbf{h}_n - \mu \nabla \xi(n) \quad (\text{A.3})$$

Le pas  $\mu$  de valeur strictement positive exprime le taux avec lequel le vecteur  $\mathbf{h}_n$  tend à minimiser l'erreur quadratique moyenne.

Evaluons à présent le vecteur gradient  $\nabla \xi(n)$ . En supposant que  $\mathbf{h}$  est complexe, le gradient est la dérivée de  $E\{|e(n)|^2\}$  par rapport à  $\mathbf{h}^*$  :

$$\nabla \xi(n) = \nabla E\{|e(n)|^2\} = E\{\nabla |e(n)|^2\} = E\{e(n) \nabla e^*(n)\} \quad (\text{A.4})$$

avec :

$$\nabla e^*(n) = -\mathbf{x}^*(n)$$

d'où :

$$\nabla \xi(n) = -E\{e(n) \mathbf{x}^*(n)\}$$

Ainsi, l'équation de mise à jour devient :

$$\mathbf{h}_{n+1} = \mathbf{h}_n - E\{e(n) \mathbf{x}^*(n)\} \quad (\text{A.5})$$

En substituant l'expression de l'erreur  $e(n)$  dans cette dernière relation, on aboutit à :

$$E\{e(n) \mathbf{x}^*(n)\} = E\{d(n) \mathbf{x}^*(n)\} - E\{\mathbf{h}_n^T \mathbf{x}(n) \mathbf{x}^*(n)\}$$

ou encore sous forme matricielle :

$$E\{e(n) \mathbf{x}^*(n)\} = \mathbf{r}_{dx} - \mathbf{R}_x \mathbf{h}_n$$

Avec :

$$\mathbf{R}_x(n) = \begin{bmatrix} E\{x(n)x^*(n)\} & E\{x(n-1)x^*(n)\} & \dots & E\{x(n-L+1)x^*(n)\} \\ E\{x(n)x^*(n-1)\} & E\{x(n-1)x^*(n-1)\} & \dots & E\{x(n-L+1)x^*(n-1)\} \\ \vdots & \vdots & \ddots & \vdots \\ E\{x(n)x^*(n-L+1)\} & E\{x(n-1)x^*(n-L+1)\} & \dots & E\{x(n-L+1)x^*(n-L+1)\} \end{bmatrix}$$

est la matrice Hermitienne d'autocorrélation de taille  $L \times L$ . On note que les coefficients de cette matrice seront constants dans le cas stationnaire.

$$\mathbf{r}_{dx} = [E\{d(n)x^*(n)\}, E\{d(n)x^*(n-1)\}, \dots, E\{d(n)x^*(n-L+1)\}]^T$$

est le vecteur d'intercorrélation entre  $d(n)$  et  $x(n)$ . Ainsi, l'algorithme du gradient stochastique devient :

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu(\mathbf{r}_{dx} - \mathbf{R}_x \mathbf{h}_n) \quad (\text{A.6})$$

Notons que si  $h_n$  est la solution des équations de Wiener-Hopf,  $h_n = R_x^{-1} r_{dx}$  [3 - 6], le terme correctif sera nul, alors  $h_{n+1} = h_n$  pour tout  $n$ . Afin de déterminer la correction pour laquelle  $h_n$  tend vers la solution de Wiener-Hopf notée  $h_{WH}$ , réécrivons l'équation (A.1) comme suit :

$$h_{n+1} = (I_L - \mu R_x) h_n + \mu r_{dx} \quad (A.7)$$

En soustrayant  $h_{WH}$  de part et d'autre de l'équation précédente, et en tenant compte du fait que  $r_{dx} = R_x h_{WH}$ , on obtient :

$$h_{n+1} - h_{WH} = (I_L - \mu R_x) h_n + \mu R_x h_{WH} - h_{WH} = [I_L - \mu R_x] (h_n - h_{WH}) \quad (A.8)$$

Définissons le vecteur d'erreur :

$$c_n = h_n - h_{WH}$$

L'équation (A.7) devient :

$$c_{n+1} = (I_L - \mu R_x) c_n \quad (A.9)$$

Notons que, si  $R_x$  n'est pas diagonale, il y aurait un inter-couplage entre les coefficients du vecteur d'erreur  $c_n$ . Cependant, nous pouvons découpler ces coefficients en diagonalisant la matrice d'autocorrélation comme suit :

$$R_x = V A V^H \quad (A.10)$$

où  $A$  est la matrice diagonale contenant les valeurs propres de  $R_x$ , et  $V$  est une matrice dont les colonnes sont les vecteurs propres de  $R_x$ . Dans la mesure où  $R_x$  est Hermitienne et définie non-négative, les valeurs propres sont réelles et non-négatives ( $\lambda_k \geq 0$ ) [3 - 6], et les vecteurs propres peuvent être choisis de manière à ce qu'ils soient orthonormaux,  $V^H V = I_L$ , i.e. :  $V$  est unitaire. En introduisant cette factorisation dans l'équation (A.9), ceci conduit à :

$$c_{n+1} = (I_L - \mu V A V^H) c_n \quad (A.11)$$

En utilisant le fait que  $V$  soit unitaire, on peut écrire l'équation précédente sous la forme :

$$c_{n+1} = (V V^H - \mu V A V^H) c_n = V (I_L - \mu A) V^H c_n$$

En multipliant de part et d'autre par  $V^H$ , on obtient :

$$V^H c_{n+1} = (I_L - \mu A) V^H c_n \quad (A.12)$$

En définissant :

$$u_n = V^H c_n$$

Alors l'équation (A.12) devient :

$$\mathbf{u}_{n+1} = (\mathbf{I}_L - \mu \mathbf{A}) \mathbf{u}_n$$

Avec un vecteur initial  $\mathbf{u}_0$ , on peut écrire :

$$\mathbf{u}_n = (\mathbf{I}_L - \mu \mathbf{A})^n \mathbf{u}_0$$

Et dans la mesure où  $(\mathbf{I}_L - \mu \mathbf{A})$  est une matrice diagonale, alors la  $k^{\text{ème}}$  composante de  $\mathbf{u}_n$  peut s'exprimer par :

$$u_n(k) = (1 - \mu \lambda_k)^n u_0(k) \quad (\text{A.13})$$

Ainsi, pour que  $\mathbf{h}_n$  converge vers  $\mathbf{h}_{WH}$ , il est nécessaire que le vecteur d'erreur  $\mathbf{c}_n$  converge vers zéro, d'où  $\mathbf{u}_n = \mathbf{V}^H \mathbf{c}_n$  converge vers zéro. Ceci se produit pour tout vecteur initial  $\mathbf{u}_0$  si et seulement si :

$$|1 - \mu \lambda_k| < 1 \quad ; \quad k = 0, 1, \dots, (L-1)$$

Ce qui conduit à choisir un pas d'adaptation  $\mu$  comme suit :

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (\text{A.14})$$

Cette condition assurera la convergence en moyenne, dans la mesure où  $E\{\mathbf{u}_n\}$  converge vers zéro. Cependant, cette majoration du pas d'adaptation est d'usage limité pour deux raisons. Premièrement, il est généralement reconnu que la limite supérieure  $2/\lambda_{\max}$  est trop grande pour assurer la stabilité de l'algorithme. Par exemple, bien que cette limitation assure que  $E\{\mathbf{h}_0\}$  converge, aucune contrainte sur la variance de  $\mathbf{h}_n$  n'est donnée. Deuxièmement, la limite supérieure de  $\mathbf{u}_n$  s'exprime en fonction de la valeur propre maximale de  $\mathbf{R}_x$ , ceci impose la connaissance au préalable de  $\mathbf{R}_x$ , ce qui n'est pas toujours le cas. Il devient alors nécessaire d'estimer  $\lambda_{\max}$ .

D'autre part, en assumant que le filtre est en régime permanent avec  $E\{\mathbf{c}_n\} = 0$ , l'erreur quadratique moyenne peut être exprimée par [3 - 6] :

$$\xi(n) = E\{|e(n)|^2\} = \xi_{\min} + \xi_{ex} \quad (\text{A.15})$$

où  $\xi_{\min} = E\{|e_{\min}(n)|^2\}$  est l'erreur quadratique moyenne minimale et  $\xi_{ex}(n)$  est l'erreur quadratique moyenne excédante qui dépend des caractéristiques statistiques de  $x(n)$ ,  $\mathbf{c}_n$  et  $d(n)$ .



En utilisant la supposition d'indépendance qui stipule que  $x(n)$  et  $h_n$  sont statistiquement indépendants, la propriété suivante peut être établie [3 - 6] :

$$\xi(\infty) = \xi_{\min} + \xi_{ex}(\infty) = \xi_{\min} \frac{1}{1 - \mu \sum_{k=0}^{L-1} \frac{\lambda_k}{2 - \mu \lambda_k}} \quad (\text{A.16})$$

et l'algorithme adaptatif converge en moyenne quadratique si et seulement si le pas d'adaptation  $\mu$  satisfait aux deux conditions suivantes :

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (\text{A.17})$$

$$\mu \sum_{k=0}^{L-1} \frac{\lambda_k}{2 - \mu \lambda_k} < 1 \quad (\text{A.18})$$

La condition (A.17) est celle qui est requise pour une convergence en moyenne et la condition (A.18) garantit que  $\xi(\infty)$  est positive. D'autre part la résolution de l'équation (A.16) donne :

$$\xi_{ex}(\infty) = \mu \xi_{\min} \frac{\sum_{k=0}^{L-1} \frac{\lambda_k}{2 - \mu \lambda_k}}{1 - \mu \sum_{k=0}^{L-1} \frac{\lambda_k}{2 - \mu \lambda_k}} \quad (\text{A.19})$$

Si on prend  $\mu \ll 2/\lambda_{\max}$ , ce qui est typiquement le cas, alors  $\mu \lambda_k \ll 2$  et la relation (A.18)

devient :

$$\frac{1}{2} \mu \sum_{k=0}^{L-1} \lambda_k < 1$$

ou bien :

$$\mu < \frac{2}{\text{tr}(\mathbf{R}_x)} \quad (\text{A.20})$$

Quand  $\mu \ll 2/\lambda_{\max}$ , il s'ensuit que :

$$\xi(\infty) \approx \xi_{\min} \frac{1}{1 - \frac{1}{2} \mu \text{tr}(\mathbf{R}_x)} \quad (\text{A.21})$$

et l'erreur quadratique moyenne excédante donnée par (A.19) est approximativement :

$$\xi_{ex}(\infty) \approx \mu \xi_{\min} \frac{\frac{1}{2} \text{tr}(\mathbf{R}_x)}{1 - \frac{1}{2} \mu \text{tr}(\mathbf{R}_x)} \approx \frac{1}{2} \mu \xi_{\min} \text{tr}(\mathbf{R}_x) \quad (\text{A.22})$$

En conséquence, pour un  $\mu$  petit,  $\xi_{ex}(\infty)$  est proportionnelle au pas d'adaptation.

## Annexe B

### Convolution rapide dans le domaine fréquentiel

Dans la première partie de cette annexe, nous présenterons quelques propriétés des matrices circulantes, utiles pour les développements ultérieurs.

#### B.1 Matrices circulantes

Une matrice  $\Psi$  de dimension  $M \times M$  est dite circulante si et seulement si :

$$\Psi_{ij} = \Psi_{1 \langle i-j \rangle M} = \psi_{\langle i-j \rangle M}, \text{ où } \langle i-j \rangle M \text{ est le résidu de } n \text{ modulo } M.$$

Les colonnes et les lignes d'une telle matrice se déduisent les unes des autres par circulation (d'où la dénomination de circulante) :

$$\Psi = \begin{bmatrix} \Psi_0 & \Psi_{M-1} & \cdots & \Psi_2 & \Psi_1 \\ \Psi_1 & \Psi_0 & \cdots & \Psi_3 & \Psi_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \Psi_{M-2} & \Psi_{M-3} & \cdots & \Psi_0 & \Psi_{M-1} \\ \Psi_{M-1} & \Psi_{M-2} & \cdots & \Psi_1 & \Psi_0 \end{bmatrix} \quad (\text{B.1})$$

De plus, toute matrice circulante  $\Psi$  peut s'écrire sous la forme [1]:

$$\Psi = W_M^{-1} X W_M \quad (\text{B.2})$$

où  $X$  est une matrice diagonale dont les éléments sont la TFD de la première colonne de  $\Psi$ .

De cette décomposition se déduisent deux propriétés :

- (1) Le produit de deux matrices circulantes est commutatif.
- (2) Le produit de deux matrices circulantes est une matrice circulante.

De l'équation (B.2) résultent tous les algorithmes rapides de convolution circulaire. En effet, par projection sur la base de Fourier, au moyen d'une transformée rapide (FFT), on transforme une convolution en produits simples.

## B.2 Convolution dans le domaine temporel

Tout algorithme de filtrage doit réaliser une convolution discrète entre une séquence d'entrée  $\{x_n\}$  de longueur infinie et une séquence  $\{h_n\}$  de longueur  $L$  finie (correspondant aux coefficients de la réponse impulsionnelle du filtre RIF).

La sortie  $y_n$  à l'instant  $n$  découle alors de la convolution discrète de la séquence  $[h_0, \dots, h_{L-1}]$  par la séquence  $[x_n, \dots, x_{n-L+1}]$  et s'exprime par :

$$y_n = \sum_{k=0}^{L-1} h_k x_{n-k} = \mathbf{h}^T \mathbf{x}_n = \mathbf{x}_n^T \mathbf{h} \quad (\text{B.3})$$

où  $\mathbf{h} = [h_0, \dots, h_{L-1}]^T$  et  $\mathbf{x}_n = [x_n, \dots, x_{n-L+1}]^T$ .

Considérons un bloc de  $N$  sorties successives évaluées sur l'intervalle  $[n, n+N-1]$ . Ce vecteur de sorties s'exprime par :

$$\begin{bmatrix} y_n \\ y_{n+1} \\ \vdots \\ y_{n+N-1} \end{bmatrix} = \begin{bmatrix} x_n & x_{n-1} & \dots & x_{n-L+1} \\ x_{n+1} & x_n & \dots & x_{n-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n+N-1} & x_{n+N-2} & \dots & x_{n+N-L} \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-1} \end{bmatrix} \quad (\text{B.4})$$

Le calcul direct de cette formulation nécessite  $NL$  multiplications et  $N(L-1)$  additions, ce qui est trop élevé notamment dans le cas d'une très longue réponse impulsionnelle.

Une extension judicieuse de la formulation (B.4) permet d'avoir une matrice circulante de taille  $N+L-1$ . De cette extension et par l'exploitation des propriétés des matrices circulantes découlent les techniques de convolution rapides OLS (*Overlap-Save* : recouvrement-sauvegarde) [17], OLA (*Overlap-Add* : recouvrement-addition) [17] et WOLA (*Weighted Overlap-Add* : Addition - recouvrement - pondéré) [30], que nous exposerons dans les prochains paragraphes.

D'une façon générale, la convolution rapide s'effectue par :

- (1) Recouvrement des entrées.
- (2) Projection sur la base de Fourier.
- (3) Multiplication terme à terme des séquences transformées.

- (4) Retour vers le domaine temporel.  
 (5) Prise en compte d'une partie du produit résultant.

### B.3 Convolution rapide par OLS

La formulation circulante s'obtient par l'extension dans le passé de la première colonne de la matrice des entrées [17], [30]. On forme alors :

$$\begin{bmatrix} \tilde{y}_{n-L+1} \\ \tilde{y}_{n-L+2} \\ \vdots \\ \tilde{y}_{n-1} \\ y_n \\ y_{n+1} \\ \vdots \\ y_{n+N-2} \\ y_{n+N-1} \end{bmatrix} = \begin{bmatrix} x_{n-L+1} & x_{n+N-1} & \dots & x_{n+N-L+1} & x_{n+N-L} & x_{n+N-L-1} & \dots & x_{n-L+2} \\ x_{n-L+2} & x_{n-L+1} & \dots & x_{n+N-L+2} & x_{n+N-L+1} & x_{n+N-L} & \dots & x_{n-L+3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n-1} & x_{n-2} & \dots & x_{n+N-1} & x_{n+N-2} & x_{n+N-3} & \dots & x_n \\ x_n & x_{n-1} & \dots & x_{n-L+1} & x_{n+N-1} & x_{n+N-2} & \dots & x_{n+1} \\ x_{n+1} & x_n & \dots & x_{n-L+2} & x_{n-L+1} & x_{n+N-1} & \dots & x_{n+2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n+N-2} & x_{n+N-3} & \dots & x_{n+N-L-1} & x_{n+N-L-2} & x_{n+N-2} & \dots & x_{n+N-1} \\ x_{n+N-1} & x_{n+N-2} & \dots & x_{n+N-L} & x_{n+N-L-1} & x_{n+N-1} & \dots & x_{n-L+1} \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-2} \\ h_{L-1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad (\text{B.5})$$

Cette dernière relation peut être réécrite de façon plus compacte :

$$\tilde{y}_n = \phi_n \tilde{h} \quad (\text{B.6})$$

Le développement de cette formulation montre clairement que les  $N$  dernières composantes du vecteur  $\tilde{y}_n$  correspondent exactement à la convolution linéaire du bloc des entrées par le filtre  $h$ , par contre les  $L-1$  premiers termes correspondent à une convolution circulaire.

En utilisant la décomposition des matrices circulantes dans le domaine fréquentiel :

$$\phi_n = W_M^{-1} \Omega_n W_M \quad (\text{B.7})$$

où  $\Omega_n$  est une matrice diagonale dont les éléments sont les coefficients de la TFD de la première colonne de la matrice  $\phi_n$  :

$$\Phi_n = W_M [x_{n-L+1}, \dots, x_{n-1}, x_n, x_{n+1}, \dots, x_{n+N-1}]^T$$

L'équation (B.6) devient :

$$\tilde{y}_n = W_M^{-1} \Phi_n \odot H \quad (\text{B.8})$$

où  $H$  est le vecteur transformé de Fourier des coefficients du filtre :

$$H = W_M \begin{bmatrix} h \\ \mathbf{0}_{(N-L+1)} \end{bmatrix} \quad (\text{B.9})$$

L'opérateur  $\odot$  représente le produit de Schur des vecteurs complexes (terme par terme). Et  $M$  la plus petite puissance de 2 supérieure ou égale à  $L+N-1$ .

L'isolement des  $N$  derniers termes de  $\tilde{y}_n$ , qui correspondent à la convolution linéaire recherchée, peut être effectuée au moyen d'une fenêtre rectangulaire  $f$  :

$$f = \begin{bmatrix} \mathbf{0}_{(L-1) \times (L-1)} & \mathbf{0}_{(L-1) \times N} \\ \mathbf{0}_{N \times L} & \mathbf{I}_{N \times N} \end{bmatrix}$$

L'algorithme de convolution rapide OLS qui en découle résulte de l'évaluation à chaque itération de l'équation :

$$y_n = f \cdot W_M^{-1} \Phi_n \odot H \quad (\text{B.10})$$

et consiste en :

- (1) L'extension du bloc des  $N$  nouvelles entrées par les  $L-1$  précédentes.
- (2) L'évaluation de  $\Phi_n$ .
- (3) La multiplication terme à terme des séquences transformées.
- (4) Le retour vers le domaine temporel par l'application de la TFD inverse.
- (5) La prise en compte uniquement des  $N$  derniers termes du vecteur résultant.

La figure (B.1) montre un schéma simplifié résumant la technique OLS.

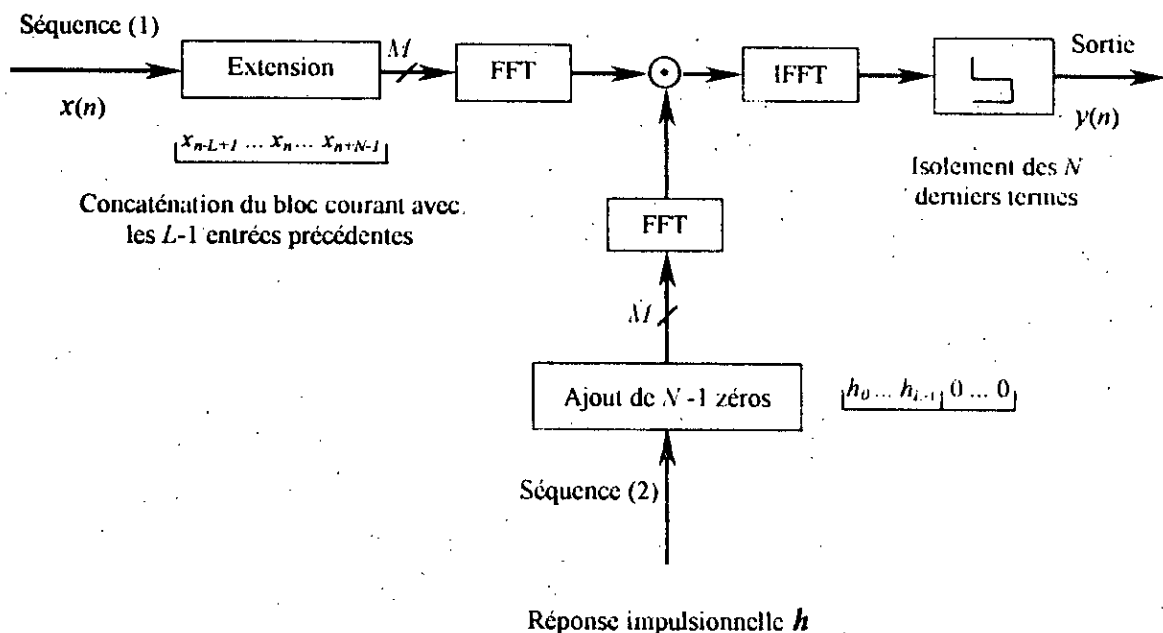


Figure B.1 : Convolution rapide par OLS.

### B.4 Convolution rapide par OLA

Dans ce cas, la taille du bloc des entrées doit être égale à la taille du filtre [17], [30]. La formulation circulante s'obtient par l'extension de la première colonne de la matrice d'entrée par  $N$  zéros. On forme alors :

$$\begin{bmatrix} \tilde{y}_{n-N} \\ \tilde{y}_{n-N+1} \\ \vdots \\ \tilde{y}_{n-1} \\ \tilde{y}_n \\ \tilde{y}_{n+1} \\ \vdots \\ \tilde{y}_{n+N-2} \\ \tilde{y}_{n+N-1} \end{bmatrix} = \begin{bmatrix} 0 & x_{n+N-1} & \dots & x_{n+1} & x_n & 0 & \dots & 0 \\ 0 & 0 & \dots & x_{n+2} & x_{n+1} & x_n & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & x_{n+N-1} & x_{n+N-2} & x_{n+N-3} & \dots & x_n \\ x_n & 0 & \dots & 0 & x_{n+N-1} & x_{n+N-2} & \dots & x_{n+1} \\ x_{n+1} & x_n & \dots & 0 & 0 & x_{n+N-1} & \dots & x_{n+2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n+N-2} & x_{n+N-3} & \dots & 0 & 0 & 0 & \dots & x_{n+N-1} \\ x_{n+N-1} & x_{n+N-2} & \dots & x_n & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-2} \\ h_{L-1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad (\text{B.11})$$

Cette formulation peut être réécrite d'une façon plus compacte :

$$\tilde{y}_n = \Psi_n \tilde{h} \quad (\text{B.12})$$

Le développement de cette égalité montre que toutes les composantes du vecteur résultant ne correspondent qu'à des convolutions partielles (sauf la dernière composante). Cependant une deuxième opération permet de retrouver la convolution linéaire désirée. Il s'agit d'effectuer un recouvrement par addition dans le domaine fréquentiel en utilisant la relation suivante [17] :

$$\Psi_n = \Psi_n' + J \odot \Psi_{n-1}' \quad (\text{B.13})$$

où  $\Psi_n'$  le vecteur transformé de Fourier de la première colonne de la matrice  $\Psi_n$  :

$$\Psi_n' = W_M [x_n, \dots, x_{n+N-1}, 0, \dots, 0]^T$$

et  $J = [J_0, \dots, J_{2N-1}]^T$ , avec :  $J_m = (-1)^m$  pour  $m = 0, \dots, 2N-1$ .

En introduisant cette relation dans l'équation (B.12) et en utilisant les propriétés des matrices circulantes, on peut écrire :

$$\tilde{y}_n = W_M^{-1} \Psi_n \odot H \quad (\text{B.14})$$

où  $H$  est défini de la même manière que dans la technique OLS (équation (B.9)). Le développement de la formulation (B.14) montre que seules les  $N$  premières composantes du vecteur  $\tilde{y}_n$  correspondent à une convolution linéaire du bloc des entrées par les coefficients du filtre, alors que les  $N$  derniers termes correspondent à une convolution circulaire.

L'isolement des  $N$  premiers termes de  $\tilde{y}_n$  correspondant à la convolution linéaire recherchée s'effectue au moyen d'une fenêtre rectangulaire  $g$  :

$$g = \begin{bmatrix} I_{N \times N} & O_{N \times (L-1)} \\ O_{(L-1) \times N} & O_{(L-1) \times (L-1)} \end{bmatrix}$$

L'algorithme de convolution rapide OLA découlant de ce qui précède consiste à évaluer à chaque itération l'équation suivante :

$$y_n = g W_M^{-1} \Psi_n \odot H \quad (\text{B.15})$$

et se résume en :

- (1) L'évaluation de  $\Psi_n$ .
- (2) Recouvrement du nouveau bloc fréquentiel avec le bloc précédent suivant la relation (B.13).
- (3) Multiplication terme à terme des séquences transformées.
- (4) Retour vers le domaine temporel par l'application de la TFD inverse.
- (5) Prise en compte uniquement des  $N$  premiers termes du vecteur résultant.

La figure B.2 montre un schéma simplifié résumant la technique OLA.

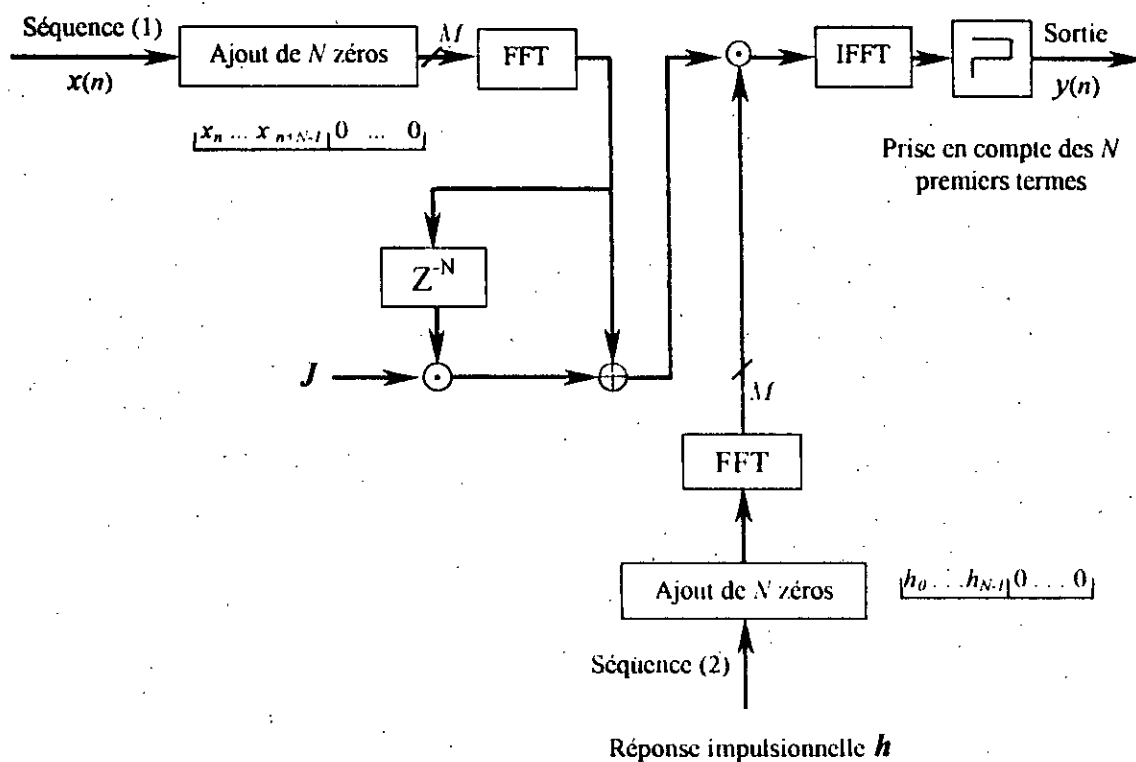


Figure B.2 : Convolution rapide par OLA

### B.5 Convolution par WOLA

Cette technique est une extension de la méthode OLS. Elle est caractérisée par la segmentation de chaque bloc d'entrée en  $\alpha$  sous blocs de taille  $R = [N/\alpha]$  ( $[x]$  signifie partie entière de  $x$ ).

A chaque nouvelle itération  $s$ , un seul nouveau sous bloc de taille  $R$  est pris en considération. Pour effectuer le calcul de la convolution [6], celui-ci sera complété par les  $(\alpha - 1)$  sous-blocs précédents afin de former un bloc de  $N$  entrées et effectuer un calcul préliminaire suivant la technique OLS. On définit une fenêtre de troncature permettant d'isoler, à chaque itération, les  $N$  dernières composantes du vecteur de sortie qui correspondent à une convolution linéaire :

$$\mathbf{f} = [0_{1 \times (\alpha-1)} \ f_0 \ \dots \ f_{N-1}]^T$$

où  $f_i \neq 0$ , pour  $i = 1, \dots, N-1$ .

Soit  $\mathbf{z}_s$  un vecteur de sortie pondéré de taille  $N$  contenant les composantes isolées par la fenêtre de reconstruction  $\mathbf{f}$  :

$$\mathbf{z}_s = [0_{N \times N} \ I_{N \times N}] (\mathbf{f} \odot \tilde{\mathbf{y}}_s)$$

où  $\tilde{\mathbf{y}}_s = \mathbf{W}_M^{-1} \Phi_s \odot \mathbf{H}$

$$\Phi_s = \mathbf{W}_M^{-1} [x_{sR-(\alpha-1)R-L+1}, \dots, x_{sR}, \dots, x_{sR+R-1}]^T$$

La particularité de cette technique relativement à la technique OLS réside dans le fait que chaque échantillon du signal de sortie peut être synthétisé à partir de  $\alpha$  blocs  $\mathbf{z}_s$  successifs, ceci en exploitant l'apparition de chaque échantillon dans  $\alpha$  blocs  $\mathbf{z}_s$  adjacents.

Une approche possible consiste à synthétiser par addition des  $\alpha$  blocs successifs un bloc de sortie de taille  $R$  dont les composantes s'obtiennent à chaque itération par :

$$y_s(n) = \sum_{k=0}^{\alpha-1} z_{s-k}(n+kR) \quad \text{pour } n = 1 : R$$

où  $y_s(j)$  et  $z_s(j)$  représentent les  $j^{\text{mes}}$  composantes des vecteurs de sortie, respectivement  $\mathbf{y}_s$  et  $\mathbf{z}_s$ , à l'itération  $s = i$ . Cette façon de synthétiser la sortie impose une condition de normalisation sur la fenêtre de reconstruction exprimée par :

$$\sum_{i=0}^{\alpha-1} f_{n-iR} = 1; \quad \text{pour } n = 1, \dots, N-1$$

La figure B.3 montre un schéma simplifié résumant la technique WOLA.



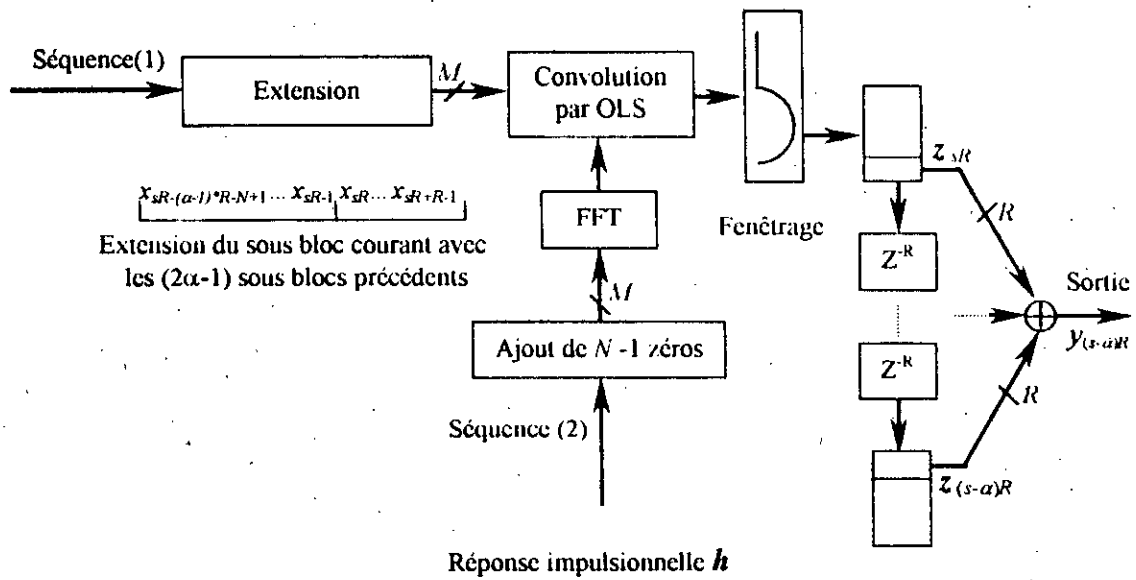


Figure B.3 : Convolution rapide par WOLA

## B.6 Etude de la complexité arithmétique

Nous développerons dans cette section les équations de complexité arithmétiques correspondants aux procédures de calcul de convolutions rapides. Le nombre d'opérations nécessaires pour le calcul d'une FFT de taille  $N$  correspond à l'algorithme Split-Radix [31]. Le nombre d'additions réelles est :

$$A_N = \frac{N}{2} (\log_2 N - 3) + 2$$

et le nombre de multiplications réelles est :

$$M_N = \frac{N}{2} (3 \log_2 N - 5) + 4$$

### B.6.1 Complexité de la procédure OLS

Le bloc des entrées est considéré de même taille que la réponse impulsionnelle. Cet algorithme utilise alors une FFT de taille  $2N$ , une FFT inverse de la même taille et  $2N$  multiplications complexes. Le nombre d'additions réelles sera donc :

$$A_{OLS} = \log_2 N - 2N + 8$$

et le nombre de multiplications réelles :

$$M_{OLS} = 2N \log_2 N + 4$$

### B.6.2 Complexité de la procédure OLA

De la même manière que pour la méthode OLS, cette procédure utilise le même nombre de multiplications et  $N$  additions supplémentaires dues au recouvrement. Le nombre d'additions est donc :

$$A_{OLA} = 6N \log_2 N - N + 8$$

et le nombre de multiplications sera :

$$M_{OLA} = 2N \log_2 N + 4$$

### B.6.3 Complexité de la procédure WOLA

De même que la méthode précédente, pour chaque nouveau bloc de taille  $R$  le nombre de multiplications reste le même. Quant aux additions la procédure WOLA nécessite  $(\alpha - 1)R$  additions supplémentaires dues aux recouvrements des sorties. Pour un traitement d'un bloc de taille  $N$  il faudra donc :

$$A_{WOLA} = \alpha(6N \log_2 N - N(1 + 1/\alpha) + 8)$$

$$M_{WOLA} = 2\alpha N \log_2 N + 4\alpha$$

### B.6.4 Complexité de la procédure à délais multiples (MDF)

Comme présentée dans le Chapitre 2, cette méthode se base sur la segmentation de la réponse impulsionnelle de taille  $L$  en  $K$  segments de taille  $N$ . On considère que la taille du bloc d'entrée est identique avec celle de chaque segment de la réponse impulsionnelle. Cette procédure nécessite en plus des deux FFT,  $2NK$  multiplications complexes et  $2N(K-1)$  additions complexes. On aura donc :

$$A_{MDF} = 6N \log_2 N + 2N(2K - 3) + 8$$

$$M_{MDF} = 2N \log_2 N + 4N(K - 1)$$

### B.6.5 Etude comparative

Nous allons établir une étude comparative entre ces différentes procédures en faisant varier la taille de la réponse impulsionnelle. La complexité est évaluée par rapport au calcul direct dans le domaine temporel, en terme de nombre d'additions réelles (figure B.4) et de multiplications réelles (figure B.5).

On remarque aisément que toutes les procédures de convolution rapide présentées offrent un gain considérable en charge de calcul, notamment pour de très longues réponses

impulsionnelles. Les techniques OLS et OLA ont approximativement la même complexité. On note également que l'utilisation d'un grand facteur de recouvrement  $\alpha$  dans la technique WOLA engendre une charge de calcul nettement plus élevée que dans le cas des techniques OLA et OLS, ceci indique que cette technique ne présente aucun intérêt en filtrage fixe. D'autre part, on constate que la segmentation de la réponse impulsionnelle augmente la complexité arithmétique.

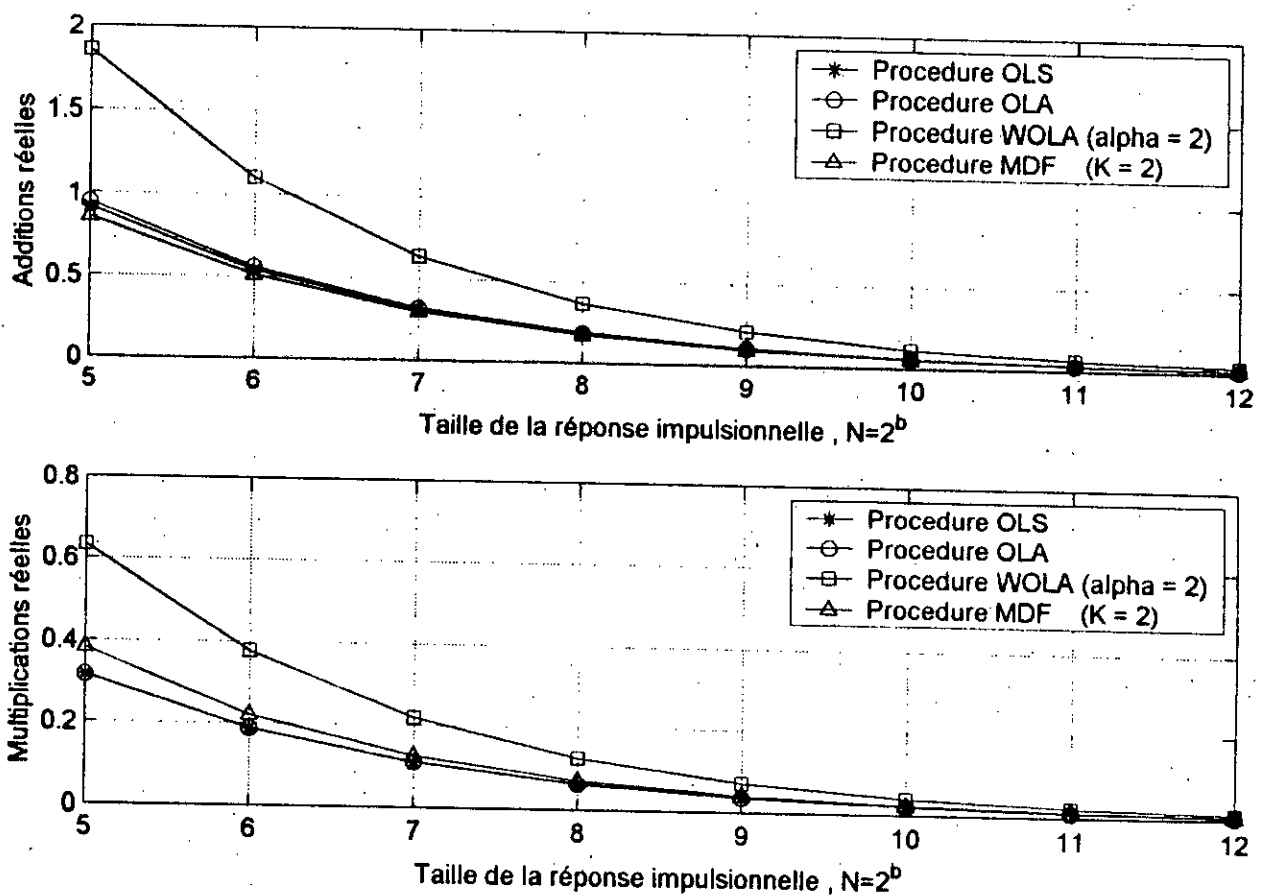


Figure B.4 : Complexité arithmétique des méthodes de convolution rapide

## Références bibliographiques

- [1] O. Ait Amrane, "*Identification de systèmes à réponse impulsionnelle longue par filtrage adaptatif en fréquence : Application à l'annulation d'écho acoustique*".  
Thèse de Doctorat, ENST, TELECOM, Paris, 1992.
- [2] S. Abdulhamid et M. Belgacem, "*Etude d'un annuleur d'écho acoustique*".  
Projet de fin d'études, ENP, Alger, 1999.
- [3] B. Widrow and S. D. Stearns, "*Adaptive Signal Processing*".  
Englewood Cliffs. NJ: Prentice-Hall, 1985.
- [4] M. H. Hayes, "*Statistical Digital Signal Processing and Modeling*".  
John Wiley & Sons, 1996.
- [5] S. Haykin, "*Adaptive Filter Theory*".  
Prentice-Hall International, Inc, 1991.
- [6] Farhang-Bouroujeny, "*Adaptive filters : Theory and applications*".  
Chichester, U. K.,: Wiley, 1998.
- [7] E. Hansler, "*The hand-free telephone problem - An annotated bibliography update*".  
Annales des Telecommunications, Vol. 49, No. 7-8, pp. 414 – 428, 1994.
- [8] P. Dreitel, E. Hansler and H. Puder, "*Acoustic echo and noise control - A long lasting challenge*".  
Damstadt University of technology, D-64283 Darsmtadt, Germany. 1999
- [9] L. Ljung, "*System identification theory for the user*".  
Prentice-Hall, Inc., New Jersey, 1987.
- [10] J. Schoukens and R. Pintelon, "*Identification of linear systems : A practical guideline to accurate modeling*".  
Perganem Press, Great Britain, 1991.

- [11] K. Eneman, M. Moonen and I. Prudler, "*Frequency domain adaptive filtering as a special case of subband echo suppression*".  
Proc. of ProRISC/IEEE Workshop on Circuits and systems, Mierlo, Nov. 1996
- [12] P. Kovintavewat, "*Modeling the impulse response of an office room*".  
Thèse de Master, Chalmers University of Technology, Goetborg, Sweden, 1998.
- [13] J. Bebesty, D. Morgan and M. Sondt, "*A better understanding and an improved solution to the problems of stereophonic acoustic echo cancellation*".  
Proc. of ICASSP-97, 1997.
- [14] J. Shynk, "*Adaptive IIR filtering*".  
IEEE ASSP magazine, Vol. 6, No. 2, pp. 4 – 21, Apr. 1989.
- [15] B. Widrow and M. Hoff, "*Adaptive Switching Circuits*".  
IREWEQCON Conv. Rec., Pt. 4, pp. 96 – 104, 1960.
- [16] G. Clarck, S. Mitra and S. Parker, "*Block implementation of adaptive digital filters*".  
IEEE Trans. on Circuits and Systems, Vol. 28, pp. 584 – 592, 1981.
- [17] J. Shynk, "*Frequency-domain and multirate adaptive filtering*".  
IEEE Signal Processing magazine, pp. 15 – 37, Jan. 1992.
- [18] M. Dentino, J. M. McCool, et B. Widrow, "*Adaptive filtering in frequency-domain*".  
Proceeding, IEEE, Vol. 66, No. 12, pp. 1658 – 1659, Dec. 1978.
- [19] D. Elliot and K. Rao, "*Fast Transforms : Algorithms, Analyses, Applications*".  
Orlando, FL : Academic, 1982.
- [20] R. Gray, "*On the Asymptotic eigenvalue distribution of Toeplitz matrices*".  
IEEE Trans. Information Theory, Vol. IT-18, No. 6, pp. 725 – 730, Nov. 1972.
- [21] M. Shensa, "*The spectral dynamics of evolving LMS adaptive filters*".  
Poceeding of IEEE Int. Conf. on ASSP, Washington, DC, pp. 950 – 953, Apr. 1979.
- [22] E. Ferrara, "*Fast implementation of LMS adaptive filters*".  
IEEE Trans. on ASSP, Vol. 28, pp. 474 – 475, 1980.
- [23] J. Prado and E. Moulines, "*Frequency-domain adaptive filtering with application to acoustic echo cancellation*".  
Annales des Telecommunications, Vol. 49, No. 7-8, pp. 414 – 428, 1994.
- [24] B. F. Boroujeny, "*Fast LMS/Newton algorithms based on Autoregressive modeling and their application to acoustic echo cancellation*".  
IEEE Trans. on ASSP, Vol. 45, No. 8, pp. 1987 – 2000, Aug. 1997.

- [25] B. F. Boroujeny and K. Chan, "*Analysis of the frequency-domain block LMS algorithm*".  
IEEE Trans. on ASSP, Vol. 48, pp.2332 – 2342, Aug. 2000.
- [26] D. Mansour and A. Gray, "*Unconstrained frequency-domain adaptive filter*".  
IEEE Trans. on ASSP, Vol. 30, pp. 726 – 734, Oct. 1982.
- [27] P. Sommen, P. Van Gerwen et H. Kotmans, "*Convergence analysis of a frequency domain adaptive filter with exponential power averaging and window function*".
- [28] J. Soo and K. Pang, "*A New structure for block FIR adaptive filtering*".  
Proceedings of IRECON, pp. 364 – 367, 1987.
- [29] J. Soo, "*Multidelay block frequency domain adaptive filters*".  
IEEE Trans. on ASSP, Vol. 38, pp.373 – 376, 1990.
- [30] E. Moulines. O. A. Amrane et Y. Grenier, "*The generalized multidelay adaptive filter: Structure and Convergence analysis*".  
IEEE Trans. on ASSP, Vol. 43, pp. 14 – 28, Jan. 1995.
- [31] P. Duhamel, "*Implementation of Split-Radix algorithms for complex, real and real-symmetric data*".  
IEEE Trans. on ASSP, Vol. 34, No. 2, pp. 285 – 295, Avr. 1986.
- [32] S. F. Boll, "*Suppression of acoustic noise in speech using spectral subtraction*".  
IEEE Trans. ASSP, Vol. 27, No. 2, Apr. 1979.
- [33] R. Martín, "*Spectral subtraction based on minimum statistics*".  
Proc. EUSIPCO-94, pp. 1182 – 1185, 1994.
- [34] C. Kermorvant, "*A comparison of noise reduction techniques for robust speech recognition*".  
IDIAP-Research Report 99-10, Jul. 1999.
- [35] H. Puder, "*Single channel noise reduction using time-frequency dependant voice activity detection*".  
Darmstadt University of technology, Merckstr. 25, 64283 Darmstadt, Germany, 1998.
- [36] P. Dreiseitel and H. Puder, "*A combination of noise reduction and improved echo cancellation*".  
Darmstadt University of technology, Merckstr. 25, 64283 Darmstadt, Germany, 1999.