

UNIVERSITE BLIDA 1

Faculté de Technologie

Département d'Electronique

THESE DE DOCTORAT

en Télécommunications

**NOUVELLES STRUCTURES ADAPTATIVES POUR LE
REHAUSSEMENT AUTOMATIQUE DU SIGNAL DE
PAROLE**

Par

Akila SAYOUD

Devant le jury composé de :

A. BENALLAL	Professeur, U. Blida 1	Président
L. MITICHE	Professeur, U. Djelfa	Examineur
T. BOUKABA	MCA, ESTA, Alger	Examineur
F. YKHLEF	MCA, U. Blida 1	Examineur
M. DJENDI	Professeur, U. Blida 1	Rapporteur
A. GUESSOUM	Professeur, U. Blida 1	Co-rapporteur

Blida, 2021

RESUME

L'objectif des travaux effectués dans le cadre de cette thèse est de développer des algorithmes efficaces pour améliorer la communication mains-libres à l'intérieur des véhicules. Dans ce contexte, cinq algorithmes bicapteurs ont été proposés pour la réduction du bruit acoustique et le rehaussement du signal de parole. Le premier et deuxième algorithmes sont basés, respectivement, sur la combinaison entre l'algorithme FNLMS (fast normalized least mean square) et la structure de séparation aveugle de sources directe et récursive. Le troisième et quatrième algorithmes reposent, respectivement, sur l'implémentation en sous-bande du premier et deuxième algorithmes. Le cinquième algorithme proposé est un algorithme à commutation pleine bande/ sous-bande, ce dernier est capable de basculer automatiquement entre les deux versions pleine bande et sous-bande de l'algorithme NLMS bicapteurs. Les algorithmes proposés ont été évalués en termes de quatre critères objectifs. Les résultats obtenus ont montré la bonne performance de ces algorithmes proposés en comparaison avec d'autres algorithmes compétitifs.

Mots clés: Filtrage adaptatif, Séparation aveugle de sources, Rehaussement de la parole, Algorithme du gradient stochastique normalisé (NLMS), Algorithme NLMS à convergence rapide, Filtrage adaptatif en sous-bande, Rapport signal à bruit, Erreur quadratique moyenne, Vitesse de convergence.

ABSTRACT

The main objective of the work reported in this thesis is to develop effective algorithms to improve hands-free communication inside vehicles. In this context, five two-sensor algorithms have been proposed for acoustic noise reduction and speech enhancement. The first and second algorithms are based, respectively, on the combination between the FNLMS (fast normalized least mean square) algorithm and the forward and backward blind source separation structure. The third and fourth algorithms are based, respectively, on the subband implementation of the first and second algorithms. The fifth algorithm is a fullband / subband switching algorithm, the latter is capable of automatically switching between the two fullband and subband versions of the NLMS two-sensor algorithm. The proposed algorithms were evaluated in terms of four objective criteria. The obtained results have shown the good performances of these proposed algorithms comparing with other competitive ones.

Keywords: Adaptive filtering, Blind source separation, Speech enhancement, Normalized least mean square algorithm (NLMS), Fast NLMS algorithm (FNLMS), Subband adaptive filtering, Signal to-noise-ratio, Mean square error, Convergence speed.

ملخص

الهدف من العمل المنفذ في إطار هذه الأطروحة هو تطوير خوارزميات فعالة لتحسين الاتصال الحر داخل المركبات. في هذا السياق، تم اقتراح خمس خوارزميات ثنائية الاستشعار لتقليل الضوضاء الصوتية وتعزيز إشارة الكلام. تستند الخوارزميات الأولى والثانية، على التوالي، إلى المزج بين خوارزمية FNLMS وهيكل فصل المصادر الأعمى المباشر والعكسي. تعتمد الخوارزميات الثالثة والرابعة، على التوالي، على تنفيذ النطاق الفرعي للخوارزميات الأولى والثانية. الخوارزمية الخامسة المقترحة هي خوارزمية تبديل بين النطاق الكامل والنطاق الفرعي، هذه الأخيرة قادرة على التبديل تلقائيًا بين إصدار النطاق الكامل وإصدار النطاق الفرعي للخوارزمية NLMS ثنائية الاستشعار. تم تقييم الخوارزميات المقترحة باستخدام أربع معايير موضوعية. أثبتت النتائج التي تم الحصول عليها الأداء الجيد لهذه الخوارزميات المقترحة مقارنة مع خوارزميات منافسة أخرى.

الكلمات المفتاحية: التصفية التكيفية، الفصل الأعمى للمصادر، الخوارزمية القياسية لأقل متوسط للمربعات (NLMS)، خوارزمية NLMS السريعة، التصفية التكيفية بحزم ثانوية، نسبة الإشارة على الضوضاء، خطأ التربيع المتوسط، سرعة التقارب.

REMERCIEMENTS

Je tiens tout d'abord à exprimer ma profonde reconnaissance et mes sincères remerciements à mon directeur de thèse, Monsieur Mohamed DJENDI, Professeur à l'université de Blida 1, pour ses conseils précieux, remarques pertinentes, et patience infinie, ainsi que le suivi intensif tout au long de ce travail de thèse, malgré ses nombreuses charges et travaux scientifiques.

J'adresse également mes remerciements à Monsieur Abderrezak GUESSOUM, Professeur à l'université de Blida 1 et responsable du laboratoire LATSI (laboratoire du traitement du signal et d'imagerie), d'avoir accepté la codirection de cette thèse.

Je tiens à remercier sincèrement Monsieur Ahmed BENALLAL, Professeur à l'université de Blida 1, d'avoir accepté de présider le jury.

Mes remerciements s'adresse aussi à Monsieur Toufik BOUKABA, Maître de conférence à l'école supérieure des techniques aéronautiques d'Alger, Monsieur Lahcen MITICHE, Professeur à l'université de Djelfa, Monsieur Farid YKHLEF, Maître de conférence à l'université de Blida 1, et qui m'ont fait l'honneur de participer au jury.

Mes derniers remerciements, reviennent à ma famille. Un grand merci à mes parents pour leur soutien. Un grand merci à tous ceux qui m'ont soutenu et supporté durant ces cinq années.

TABLE DES MATIERES

RESUME	01
ABSTRACT	02
ملخص	03
REMERCIEMENTS	04
TABLE DES MATIERES	05
LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX	07
INTRODUCTION GENERALE	15
1. GENERALITES SUR LE FILTRAGE ET ALGORITHMES ADAPTATIFS	17
1. 1. Introduction	17
1. 2. Filtrage de Wiener	17
1. 3. Filtrage adaptatif	19
1. 4. Présentation des algorithmes adaptatifs	21
1. 5. Critères de comparaison et choix des algorithmes adaptatifs	29
1. 6. Conclusion	29
2. GENERALITES SUR LES METHODES DE REDUCTION DU BRUIT ACOUSTIQUE ET LE REHAUSSEMENT DU SIGNAL DE PAROLE.	30
2. 1. Introduction	30
2. 2. Réduction du bruit acoustique	30
2. 3. Réduction du bruit acoustique par la séparation aveugle de sources	36
2. 4. Conclusion	44
3. ALGORITHMES BICAPTEURS POUR LE REHAUSSEMENT DU SIGNAL DE PAROLE	45
3. 1. Introduction	45

3. 2. Position du problème	46
3. 3. Présentation des algorithmes bicapteurs proposés	47
3. 4. Initialisation des algorithmes proposés	56
3. 5. Evaluation de la complexité de calcul	56
3. 6. Résultats de simulation des algorithmes proposés	57
3. 7. Conclusion	80
4. NOUVEAUX ALGORITHMES BICAPTEURS EN SOUS- BANDE	81
4. 1. Introduction	81
4. 2. Filtrage adaptatif en sous-bande	81
4. 3. Algorithmes bicapteurs en sous-bande basés sur la BSS	84
4. 4. Evaluation de la complexité de calcul	98
4. 5. Résultats de simulation des algorithmes bicapteurs en sous-bande	98
4. 6. Conclusion	114
5. NOUVEL ALGORITHME BICAPTEURS A COMMUTATION PLEINE BANDE/SOUS-BANDE	115
5. 1. Introduction	115
5. 2. Algorithme à commutation pleine bande/ sous-bandes proposé	115
5. 3. Evaluation de l'algorithme proposé	117
5. 4. Conclusion	125
CONCLUSION GENERALE	127
APPENDICES	129
A. Liste des symboles et des abréviations	129
REFERENCES	131

LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX

Liste des figures

Chapitre 1

Figure 1. 1	: Représentation schématique d'un filtre de Wiener.	17
Figure 1. 2	: Schéma du principe de base du filtrage adaptatif.	19
Figure 1. 3	: Principe de l'identification des systèmes.	19
Figure 1. 4	: Principe de la prédiction.	20
Figure 1. 5	: Principe d'annulation d'interférence.	21

Chapitre 2

Figure 2. 1	: Catégorisation des méthodes de réduction du bruit.	31
Figure 2. 2	: Modèle de débruitage mono-capteur.	31
Figure 2. 3	: Modèle de débruitage multi-capteurs.	34
Figure 2. 4	: Structure d'annulation adaptative de bruit.	35
Figure 2.5	: Structure générale de la séparation aveugle de sources.	37
Figure 2.6	: Modèle général du mélange convolutif.	40
Figure 2.7	: Modèle simplifié du mélange convolutif.	41
Figure 2.8	: Structure de séparation aveugle de sources directe (FBSS).	42
Figure 2.9	: Structure de séparation aveugle de sources récursive (BBSS).	43

Chapitre 3

Figure 3.1	: Schéma du modèle de mélange convolutif simplifié adopté.	46
Figure 3.2	: Schéma détaillé de l'Algorithme 1	47
Figure 3.3	: Schéma détaillé de l'Algorithme 2.	52
Figure 3.4	: Signal de parole $s(n)$ avec son spectrogramme, $f_e = 8kHz$.	58
Figure 3.5	: Signal de bruit USASI avec son spectrogramme, $f_e = 8kHz$.	58

- Figure 3.6 : Exemple d'une réponse implusionnelle simulée $h_{21}(n)$, avec $L = 128$. 60
- Figure 3.7 : Exemple d'une réponse implusionnelle simulée $h_{12}(n)$, avec $L = 128$. 60
- Figure 3.8 : Signal observé $p_1(n)$ avec son spectrogramme. 61
- Figure 3.9 : Signal observé $p_2(n)$ avec son spectrogramme. 61
- Figure 3.10 : Signal de parole originale $s(n)$ avec sa segmentation (DAV) manuelle. 62
- Figure 3.11 : Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par NLMS 1, (c) signal de sortie obtenu par l'Algorithme 1. 66
- Figure 3.12 : Evaluation du critère RSB segmental de l'algorithme 1 proposé et l'algorithme NLMS 1, pour un bruit blanc à l'entrée, avec deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 67
- Figure 3.13 : Evaluation du critère RSB segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 67
- Figure 3.14 : Evaluation du critère de désajustement pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 69
- Figure 3.15 : Evaluation du critère de désajustement pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 69
- Figure 3.16 : Evaluation du critère EQM segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 = RSB_2 = 3dB$. 70

- Figure 3.17 : Evaluation du critère EQM segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 = RSB_2 = 3dB$. 70
- Figure 3.18 : Evaluation du critère de la DC pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 72
- Figure 3.19 : Evaluation du critère de la DC pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 72
- Figure 3.20 : Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par NLMS 2, (c) signal de sortie obtenu par l'Algorithme 2. 74
- Figure 3.21 : Evaluation du critère RSB segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 75
- Figure 3.22 : Evaluation du critère RSB segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 75
- Figure 3.23 : Evaluation du critère de désajustement pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 76
- Figure 3.24 : Evaluation du critère de désajustement de l'algorithme 2 proposé et l'algorithme NLMS 2, pour un bruit USASI à l'entrée, avec deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 = RSB_2 = 3dB$. 76

- Figure 3.25 : Evaluation du critère EQM segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 78
- Figure 3.26 : Evaluation du critère EQM segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 78
- Figure 3.27 : Evaluation du critère de la DC pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 79
- Figure 3.28 : Evaluation du critère de la DC pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$. 79

Chapitre 4

- Figure 4.1 : Structure générale du filtrage adaptatif en sous-bande. 82
- Figure 4.2 : Opération de décimation par un facteur D . 83
- Figure 4.3 : Opération d'interpolation par un facteur I . 83
- Figure 4.4 : Banc de filtres à M canaux (a) banc d'analyse (b) banc de synthèse. 83
- Figure 4.5 : Schéma détaillé de l'Algorithme 3. 85
- Figure 4.6 : Schéma détaillé du processus d'adaptation de l'Algorithme 3. 86
- Figure 4.7 : Schéma détaillé de l'Algorithme 4. 91
- Figure 4.8 : Schéma détaillé du processus d'adaptation de l'Algorithme 4. 93
- Figure 4.9 : Modèle de mélange. 98
- Figure 4.10 : Caractéristique fréquentielle des filtres d'analyse et de synthèse: [en haut] pour 2 sous-bandes et [en bas] pour 4 sous-bandes. 99

- Figure 4.11 : Evolution temporelle du: (a) signal de parole originale, (b) 103
signal de sortie obtenu par l'Algorithme 1, (c) signal de sortie
obtenu par l'Algorithme 3 avec $M = 2$, (d) signal de sortie
obtenu avec l'Algorithme 3 avec $M = 4$.
- Figure 4.12 : Evaluation du critère de désajustement de l'algorithme 3 104
proposé ($M = 2$ et 4) et l'algorithme 1, pour deux niveaux de
RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.
- Figure 4.13 : Evaluation du critère d'EQM segmental de l'algorithme 3 105
proposé (avec $M = 2$ et 4) et l'algorithme 1, pour deux
niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à
droite] $RSB_1 = RSB_2 = 3dB$.
- Figure 4.14 : Evaluation du critère RSB segmental de l'algorithme 3 106
proposé (avec $M = 2$ et 4) et l'algorithme 1, pour deux
niveaux de RSB d'entrée: [à gauche], $RSB_1 = RSB_2 = 0$, [à
droite] $RSB_1 = RSB_2 = 3dB$.
- Figure 4.15 : Evaluation du critère de la DC de l'algorithme 3 proposé (avec 107
 $M = 2$ et 4) et l'algorithme 1, pour deux niveaux de RSB
d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.
- Figure 4.16 : Evolution temporelle du: (a) signal de parole originale, (b) 110
signal de sortie obtenu par l'Algorithme 2, (c) signal de sortie
obtenu par l'Algorithme 4 avec $M = 2$, (d) signal de sortie
obtenu avec l'Algorithme 4 avec $M = 4$.
- Figure 4.17 : Evaluation du critère de désajustement de l'algorithme 4 111
proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux
niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$ et [à
droite] $RSB_1 = RSB_2 = 3dB$.
- Figure 4.18 : Evaluation du critère d'EQM segmental de l'algorithme 4 112
proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux
de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$ et [à droite]

$$RSB_1 = RSB_2 = 3dB.$$

Figure 4.19 : Evaluation du critère RSB segmental de l'algorithme 4 113
proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux
de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$ et [à droite]
 $RSB_1 = RSB_2 = 3dB$.

Figure 4.20 : Evaluation du critère de la DC de l'algorithme 4 proposé (avec 114
 $M = 2$ et 4) et l'algorithme 2, pour deux niveaux de RSB
d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$ et [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.

Chapitre 5

Figure 5.1 : Schéma bloc de l'algorithme proposé (Algorithme 5). 116

Figure 5.2 : Evolution temporelle du: (a) signal de parole original, (b) 119
signal de sortie obtenu par l'algorithme NLMS 1, (c) signal de
sortite obtenu par l'algorithme SNLMS avec $M = 2$, (d) signal
de sortie obtenu par l'Algorithme 5 avec $M = 2$.

Figure 5.3 : Evaluation du critère de désajustement de l'algorithme 5 121
proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$) ,
pour un bruit USASI à l'entrée, et deux niveaux de RSB
d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.

Figure 5.4 : Evaluation du critère de désajustement de l'algorithme 5 121
proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$) ,
pour un bruit USASI à l'entrée, et deux niveaux de RSB
d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.

Figure 5.5 : Evaluation du critère d'EQM Segmentale de l'algorithme 5 122
proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$) ,
pour un bruit USASI à l'entrée, et deux niveaux de RSB
d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 =$
 $RSB_2 = 3dB$.

- Figure 5.6 : Evaluation du critère d'EQM Segmentale de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 = RSB_2 = 3dB$. 122
- Figure 5.7 : Evaluation du RSB Segmental de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 = RSB_2 = 3dB$. 124
- Figure 5.8 : Evaluation du RSB Segmental de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 = RSB_2 = 3dB$. 124
- Figure 5.9 : Evaluation de la DC de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 = RSB_2 = 3dB$. 125
- Figure 5.10 : Evaluation de la DC de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ et [à droite] $RSB_1 = RSB_2 = 3dB$. 125

Liste des tableaux

Chapitre 1

Tableau 1.1	Complexité de calcul de l'algorithme LMS.	23
Tableau 1.2	Complexité de calcul de l'algorithme NLMS.	24
Tableau 1.3	Complexité de calcul de l'algorithme FNLMS.	28

Chapitre 3

Tableau 3.1	Listing du premier algorithme proposé (Algorithme 1).	51
Tableau 3.2	Listing du deuxième algorithme proposé (Algorithme 2).	55

Tableau 3.3 Initialisation des algorithmes proposés: Algorithme1 et Algorithme 2. 56

Tableau 3.4 Complexité de calcul des deux algorithmes (Algorithme 1 et Algorithme 2). 57

Tableau 3.5 Paramètres de simulation de l'Algorithme 1 et NLMS 1. 65

Tableau 3.6 Paramètres de simulation de l'Algorithme 2 et NLMS 2. 73

Chapitre 4

Tableau 4.1 Résumé de l'Algorithme 3. 90

Tableau 4.2 Résumé de l'Algorithme 4. 97

Tableau 4.3 Complexité de calcul des deux algorithmes (Algorithme 3 et Algorithme 4). 98

Tableau 4.4 Paramètres de simulation de l'Algorithme 1 et l'Algorithme 3. 101

Tableau 4.5 Paramètres de simulation de l'Algorithme 2 et l'Algorithme 4. 108

Chapitre 5

Tableau 5.1 Paramètres de simulation de NLMS 1, SNLMS et Algorithme 5 118

INTRODUCTION GENERALE

Dans le contexte des télécommunications modernes, les systèmes de communication mains-libres sont sans doute le moyen de communication le plus populaire et le plus répandu de nos jours, ce qui s'explique par la flexibilité et la mobilité accrues qu'ils offrent, ainsi que par les exigences de sécurité et de confort lors de son utilisation en véhicule. La voiture s'avère être un environnement acoustique particulièrement bruyant. En effet, le bruit vient de multiples sources, comme le moteur, le roulement des pneus sur la route, Ces sources varient fortement d'une condition de conduite à l'autre. La réduction du bruit au niveau de la prise de son dans l'habitacle constitue donc un élément majeur dans le confort des utilisateurs de la téléphonie mains-libres [1].

La réduction du bruit acoustique a pour but, en premier lieu, d'améliorer l'intelligibilité de la conversation pour qu'elle soit fluide, mais aussi de conserver une voie naturelle, pour le confort du locuteur distant [1]. De nombreuses techniques de réduction du bruit acoustique et de rehaussement de la parole ont été proposées dans la littérature [2-4]. Généralement la classification de ces techniques repose sur le nombre de microphones disponible pour le traitement, où, on peut distinguer deux catégories de base: techniques de réduction du bruit mono-capteur [5-6] et multi-capteurs [7-8].

Récemment, une attention particulière a été portée aux méthodes de réduction du bruit bicapteurs, notamment, les deux structures de séparation aveugle de sources directe [9] et récursive [10]. De nombreux chercheurs ont suggérés la combinaison de ces deux structures avec les algorithmes du filtrage adaptatif dans différents domaines: temporel [11-12] et fréquentiel [13-14]. Ces deux structures ont donné un nouvel aperçu du domaine de la réduction du bruit acoustique et le rehaussement du signal de parole, c'est pourquoi elles feront l'objet de notre travail.

L'objectif des travaux effectués dans le cadre de cette thèse est de développer des algorithmes bicapteurs efficaces pour améliorer la réduction du bruit acoustique dans les systèmes de communication mains-libres, en particulier

la téléphonie mains-libres à l'intérieur des véhicules. Dans ce contexte, nous avons proposé cinq algorithmes bicapteurs basés sur les deux structures de séparation aveugle de sources directe et récursive dans le but: d'améliorer la vitesse de convergence, minimiser la distorsion du signal de parole estimé. Sans perte de généralité, nous présenterons ci-après un aperçu des différents chapitres de cette thèse:

➤ Le premier chapitre de cette thèse, est consacré au filtrage adaptatif, il présente la structure d'un filtre adaptatif, les différentes applications et les algorithmes du filtrage adaptatif.

➤ Le second chapitre présente, un état de l'art des techniques mono-capteurs, bicapteurs et multi-capteurs les plus utilisées dans le domaine de débruitage de la parole.

➤ Le troisième chapitre, est consacré au développement de deux algorithmes bicapteurs, basés sur la combinaison des deux structures de séparation aveugle de sources directe et récursive avec l'algorithme du filtrage adaptatif FNLMS (fast normalized least mean square). Une étude théorique de ces deux algorithmes ainsi que leurs résultats de simulation seront détaillés et présentés.

➤ Le quatrième chapitre présente, deux nouveaux algorithmes bicapteurs en sous-bande. Ces deux algorithmes reposent sur l'implémentation en sous-bande des deux algorithmes présentés au chapitre 3. Il présente aussi la validation expérimentale et les résultats d'une étude comparative entre chaque algorithme en sous-bande proposé et sa version pleine bande.

➤ Le cinquième chapitre, présente un algorithme à commutation pleine bande/ sous-bande, dans le but d'améliorer le comportement de l'algorithme NLMS bicapteurs en sous-bandes. Il présente aussi les résultats de simulation, les comparaisons, et les discussions.

Enfin, nous terminerons cette thèse par une conclusion générale, avec une présentation des perspectives pour des travaux futurs envisagés dans la continuité de nos travaux de recherche.

CHAPITRE 1

GENERALITES SUR LE FILTRAGE ET ALGORITHMES ADAPTATIFS

1.1. Introduction

Le présent chapitre présentera les concepts théoriques sur lesquelles reposent les techniques du filtrage optimal et adaptatif. Ensuite les principaux algorithmes du filtrage adaptatif seront discutés, en particulier les algorithmes du type gradient stochastique LMS (least mean square), et sa version normalisé NLMS (normalized least mean square), l'algorithme NLMS à convergence rapide FNLMS (fast normalized least mean square).

1.2. Filtrage de Wiener

Le filtre de Wiener a été introduit par Norbert Wiener dans les années 1940. La théorie de Wiener délivre le filtre qui minimise l'erreur quadratique moyenne (EQM) entre sa sortie et une séquence désirée [15]. La figure 1.1 montre une représentation schématique d'un filtre de Wiener w . Le filtre à comme entrée un signal de référence $x(n)$ et produit un signal de sortie $y(n)$, où $y(n)$ est l'estimation au sens de l'EQM minimal du signal désiré $d(n)$.

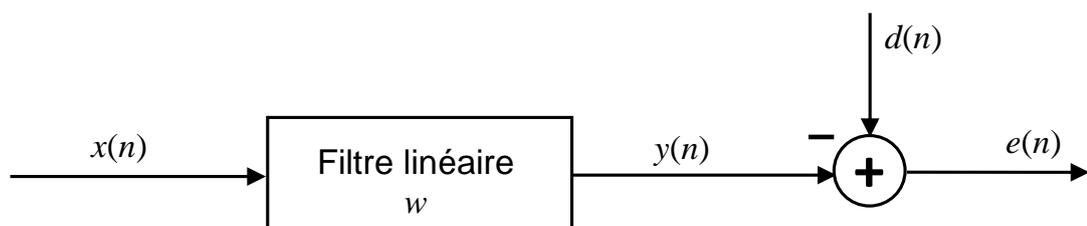


Figure 1. 1 : Représentation schématique d'un filtre de Wiener.

Le signal de sortie $y(n)$ du filtre w est donné par:

$$y(n) = \sum_{m=0}^{L-1} w_m x(n - m) \quad (1.1)$$

Ce dernier peut encore s'écrire, en introduisant la notation vectorielle :

$$y(n) = \mathbf{w}^T \mathbf{x}(n) \quad (1.2)$$

Où $\mathbf{w} = [w_0, w_1, \dots, \dots, w_{L-1}]^T$ représente le vecteur de L coefficient du filtre de Wiener, et $\mathbf{x}(n) = [x(n), x(n - 1), \dots, \dots, x(n - L + 1)]^T$ est le vecteur qui résume le

passé du signal d'entrée $x(n)$ sur une durée de L échantillons. $(.)^T$ désigne l'opérateur de la transposition.

Le signal d'erreur du filtre de Wiener est défini comme la différence entre le signal désiré $d(n)$ et le signal de sortie du filtre $y(n)$:

$$e(n) = d(n) - y(n) = d(n) - \mathbf{w}^T \mathbf{x}(n) \quad (1.3)$$

Plus l'erreur est faible, plus l'estimation sera bonne. Dans le filtre de Wiener, la fonction coût, dénommée J , qui sera à minimiser est l'erreur quadratique moyenne (EQM) définie par:

$$\begin{aligned} J &= E[e^2(n)] \\ &= \sigma_d^2 - 2\mathbf{r}_{xd}\mathbf{w}^T + \mathbf{w}^T \mathbf{R}_{xx} \mathbf{w} \end{aligned} \quad (1.4)$$

Où

$\sigma_d^2 = E[d^2(n)]$ est la variance du signal désiré.

$\mathbf{r}_{xd} = E[\mathbf{x}(n)d(n)]$ est le vecteur d'inter-corrélation entre le signal désiré et le signal d'entrée.

$\mathbf{R}_{xx} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ est la matrice d'auto-corrélation du signal d'entrée $\mathbf{x}(n)$.

$(.)^T$ désigne l'opérateur de la transposition.

$E[.]$ désigne la moyenne statistique.

Les coefficients du filtre optimal \mathbf{w}_{opt} sont obtenus en annulant le gradient de l'erreur quadratique moyenne par rapport aux coefficients du filtre:

$$\begin{aligned} \nabla J &= \frac{\partial J}{\partial \mathbf{w}} \\ &= -2E[e(n)\mathbf{x}(n)] \end{aligned} \quad (1.5)$$

Où, ∇ désigne l'opérateur gradient.

Pour $\nabla J = 0$, on aura à l'optimum:

$$-2E[e_{min}\mathbf{x}(n)] = 0 \quad (1.6)$$

Où, e_{min} est l'erreur pour laquelle la fonction coût J est minimisée. C'est le principe d'orthogonalité signifiant que toutes les entrées $\mathbf{x}(n)$ sont décorrélatées de l'erreur e_{min} [16].

En développant cette dernière équation on obtient:

$$E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{w}_{opt} = E[\mathbf{x}(n)d(n)] \quad (1.7)$$

$$\mathbf{R}_{xx}\mathbf{w}_{opt} = \mathbf{r}_{xd} \quad (1.8)$$

Cette dernière relation (1.8) est appelée équation de Wiener-Hopf. La solution est donnée par l'équation suivante:

$$\mathbf{w}_{opt} = \mathbf{R}_{xx}^{-1} \mathbf{r}_{xd} \quad (1.9)$$

La mise en œuvre d'un filtre optimal de Wiener nécessite la connaissance des caractéristiques des signaux d'entrée. Cependant, dans la plupart des applications pratiques, les statistiques des signaux sont inconnues à priori où les signaux sont souvent non-stationnaires. Par conséquent, les statistiques doivent être évaluées à partir des données recueillies aux entrées. Ce type de problème amène à développer des méthodes de filtrage adaptatif où les coefficients du filtre sont ajustés à chaque instant n par un algorithme d'adaptation itératif [17].

1.3. Filtrage adaptatif

1.3.1. Principe de filtrage adaptatif

Le principe du filtrage adaptatif est représenté par la figure 1.2, il repose classiquement sur deux étapes [18] :

- Une étape de filtrage qui permet de fournir un signal de sortie $y(n)$ en convoluant le signal d'entrée $x(n)$ avec les coefficients du filtre adaptatif $w(n)$.
- Une étape d'adaptation qui permet d'ajuster les coefficients du filtre adaptatif $w(n)$ suivant un algorithme donné. L'algorithme du filtrage adaptatif permet de calculer les coefficients du filtre $w(n)$ de façon à ce que le signal d'erreur $e(n) = d(n) - y(n)$ soit minimisé au sens d'un critère préalablement défini. Le signal d'erreur $e(n)$ diminue jusqu'à atteindre dans certains cas une valeur nulle. Lorsque la valeur est atteinte, les coefficients du filtre adaptatif cessent de s'adapter.

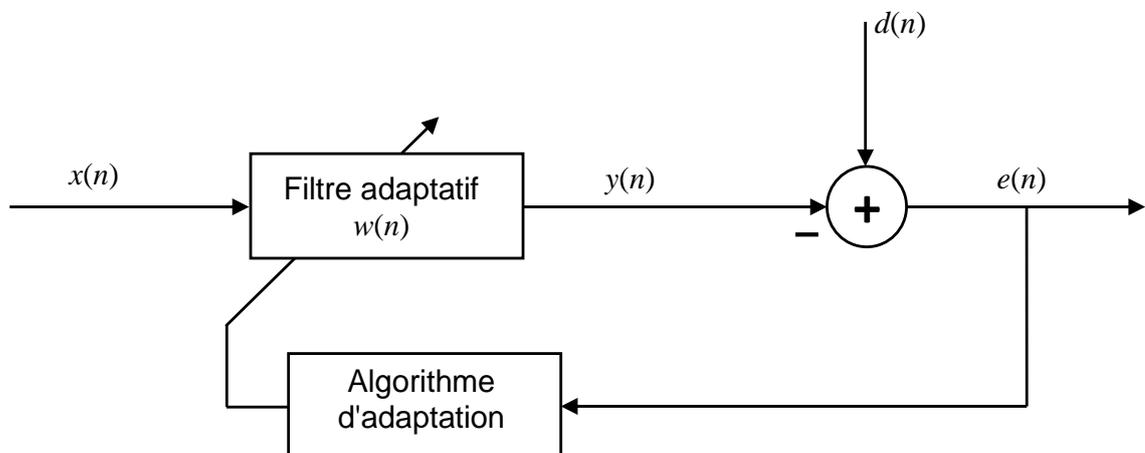


Figure 1. 2 : Schéma de principe de base du filtrage adaptatif.

1.3.2. Applications du filtrage adaptatif

Les filtres adaptatifs ont été appliqués avec succès dans divers domaines, souvent de nature très différente. La différence essentielle réside dans la manière dont la réponse souhaitée est obtenue. Ci-dessous, on présente trois classes d'applications du filtrage adaptatif :

- *Identification des systèmes*

L'identification du système est une approche permettant de modéliser un système inconnu. Dans cette configuration, le système inconnu est en parallèle avec un filtre adaptatif et les deux sont excités avec le même signal ($x(n)$). Le filtre représentant le modèle sera estimé à partir de l'observation de la différence entre la sortie du système et son estimation à la sortie du filtre. La figure ci-dessous représente le principe de l'identification, dont $d(n)$ est la sortie du système que l'on souhaite identifié [19].

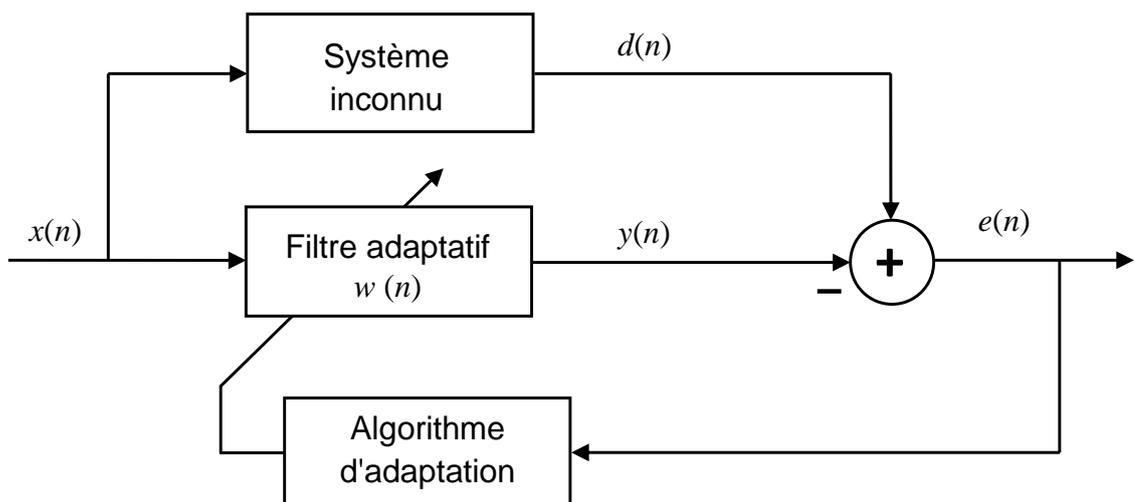


Figure 1. 3 : Principe de l'identification des systèmes.

- *Prédiction*

La prédiction consiste en l'estimation de la valeur future d'un signal à partir de l'information passée, où le signal d'entrée $x(n)$ du filtre adaptatif est une version retardée du signal désiré $d(n)$. La sortie du filtre adaptatif $y(n)$ est le signal prédit à partir des valeurs précédentes du signal désiré [20]. Le principe de la prédiction est schématisé par la figure 1.4.

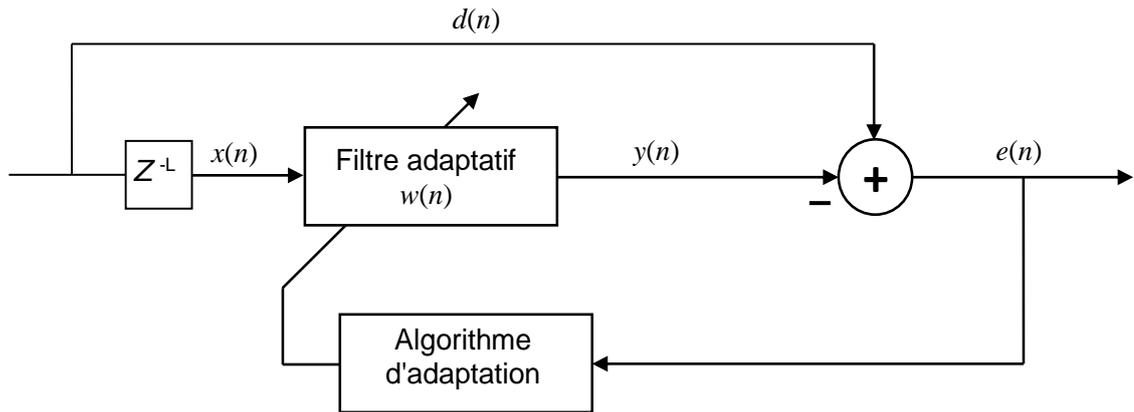


Figure 1. 4 : Principe de la prédiction.

- *Annulation d'interférence*

Dans cette dernière classe d'application, le filtre adaptatif est utilisé pour annuler les interférences inconnues contenues dans un signal primaire. Le principe d'annulation d'interférence est présenté par la figure 1.5, où $d(n)$ est un signal primaire. Il se compose d'un signal utile, non observé que l'on souhaite estimer, pollué de manière additive par un bruit supposé indépendant du signal utile. $x(n)$ est un signal de référence, dénué (ou presque) d'information et obtenu par un capteur proche des interférences. L'objectif est d'estimer le bruit qui entache le signal utile via un filtre, puis par la suite le soustraire du signal primaire pour acquérir à la fin le signal utile tout seul [21].

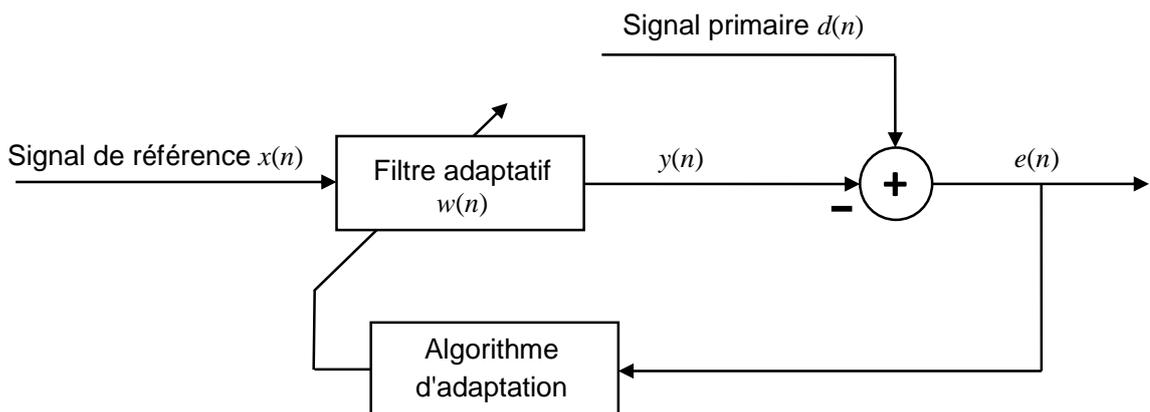


Figure 1. 5 : Principe d'annulation d'interférence.

1.4. Présentation des algorithmes adaptatifs

Les algorithmes du filtrage adaptatif associés à la structure transverse RIF (réponse impulsionnelle finie) sont très nombreux et ont largement été étudiés

dans la littérature [22]. Nous distinguons deux familles principales: la famille des gradients stochastique LMS (least mean square) et celle des moindres carrés récursifs RLS [23] (recursive least squares). La plupart des autres algorithmes découlent de ces deux derniers. Dans les paragraphes qui suivent nous allons présenter les algorithmes adaptatifs qui seront utilisés dans le reste de ce manuscrite à savoir : algorithmes du type gradient stochastique LMS et NLMS et l'algorithme NLMS à convergence rapide FNLMS (fast normalized least mean square).

1.4.1. Algorithme du gradient stochastique LMS

L'algorithme LMS (least mean square) est une approximation stochastique de l'algorithme du gradient déterministe. Cet algorithme est le plus populaire en raison de son extrême simplicité de mise en œuvre et sa robustesse [17].

L'algorithme du gradient déterministe est défini par [24] :

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu[\mathbf{r}_{xd} - \mathbf{R}_{xx}\mathbf{w}(n-1)] \quad (1.10)$$

Où, μ est un pas d'adaptation qui contrôle la convergence et la stabilité de l'algorithme. Le problème majeur dans l'algorithme du gradient déterministe est que le vecteur d'inter-corrélation $\mathbf{r}_{xd} = E[\mathbf{x}(n)d(n)]$ et la matrice d'auto-corrélation $\mathbf{R}_{xx} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ sont évidemment inconnus.

L'idée de base des algorithmes de type gradient stochastique est de remplacer ces grandeurs déterministes par des estimées $\hat{\mathbf{R}}_{xx}$ et $\hat{\mathbf{r}}_{xd}$ à l'instant n . Dans le cas de l'algorithme LMS, on choisit les estimées les plus simples possibles [25], à savoir:

$$\hat{\mathbf{R}}_{xx} = \mathbf{x}(n)\mathbf{x}^T(n) \quad (1.11)$$

$$\hat{\mathbf{r}}_{xd} = \mathbf{x}(n)d(n) \quad (1.12)$$

En remplaçant (1.11) et (1.12) dans l'expression du gradient déterministe qui est donnée par (1.10), on obtient l'algorithme du gradient stochastique LMS:

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu e(n)\mathbf{x}(n) \quad (1.13)$$

Avec $e(n)$ est l'erreur de filtrage a priori donnée par:

$$e(n) = d(n) - \mathbf{w}^T(n-1)\mathbf{x}(n) \quad (1.14)$$

D'après [26], la condition nécessaire et suffisante pour la convergence de l'algorithme LMS est:

$$0 < \mu < \frac{2}{\lambda_{max}} \quad (1.15)$$

Où λ_{max} est la plus grande valeur propre de la matrice d'auto-corrélation \mathbf{R}_{xx} du signal d'entrée.

En pratique, on utilise souvent, pour le choix de la constante μ , une condition plus restrictive :

$$0 < \mu \leq \frac{2}{\text{Trace}(\mathbf{R}_{xx})} = \frac{2}{L\sigma_x^2} \quad (1.16)$$

Où $\text{Trace}(\mathbf{R}_{xx})$ représente la somme des éléments de la diagonale de la matrice \mathbf{R}_{xx} et σ_x^2 désigne l'énergie du signal d'entrée $x(n)$.

Les principaux atouts de l'algorithme LMS sont sa faible complexité numérique, sa simplicité d'implémentation et sa robustesse aux erreurs, par contre, sa vitesse de convergence dépend des statistiques du signal d'entrée et se dégrade rapidement lorsque des signaux fortement corrélés (tels que des signaux de parole) sont utilisés [18].

- *Complexité de calcul*

La complexité algorithmique de l'algorithme LMS est connue: chaque itération n comporte $(2L + 1)$ multiplications [18]. Le tableau 1.1 résume les opérations nécessaires pour l'algorithme LMS à chaque itération n .

Tableau 1.1 : Complexité de calcul de l'algorithme LMS.

Etapes de l'algorithme	Nombre de Mult/Div
Etape de filtrage:	
$e(n) = d(n) - \mathbf{w}^T(n-1)\mathbf{x}(n)$	L
Etape d'adaptation:	
$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu e(n)\mathbf{x}(n)$	$L + 1$
Complexité de calcul totale	$2L + 1$

1.4.2. Algorithme du gradient stochastique normalisé (NLMS)

L'algorithme NLMS (normalized least mean square) consiste à normaliser le pas d'adaptation μ dans l'algorithme LMS par rapport à l'énergie du signal d'entrée pour réduire au minimum l'effet de la variation de la puissance du signal d'entrée

et de rendre ainsi la convergence plus au moins uniforme en passant d'une étape d'adaptation à une autre [27].

Le pas d'adaptation μ est alors remplacé par un pas d'adaptation μ_n défini à chaque itération par:

$$\mu_n = \frac{\mu}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (1.17)$$

La mise à jour des coefficients du filtre adaptatif par l'algorithme NLMS est donnée par:

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \frac{\mu}{\mathbf{x}^T(n)\mathbf{x}(n) + \epsilon} \mathbf{x}(n)e(n) \quad (1.18)$$

Où ϵ est un facteur permettant de suivre plus ou moins rapidement les variations d'énergie dans le signal d'entrée $x(n)$.

- *Complexité de calcul*

La complexité de calcul de l'algorithme NLMS est de l'ordre de $3L + 2$ multiplications par itération. Le tableau 1.2 résume les opérations nécessaires pour l'algorithme NLMS à chaque itération n .

Tableau 1.2 : Complexité de calcul de l'algorithme NLMS.

Etape de l'algorithme	Nombre de Mult/Div
Etape de filtrage:	
$e(n) = d(n) - \mathbf{w}^T(n-1)\mathbf{x}(n)$	L
Etape d'adaptation:	
$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu \frac{e(n)\mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{x}(n) + \epsilon}$	$2L + 2$
Complexité de calcul totale	$3L + 2$

1.4.3. Algorithme NLMS à convergence rapide (FNLMS)

L'algorithme FNLMS (fast normalized least mean square) est récemment conçu pour les applications d'annulation d'écho [28]. Ce dernier est obtenu à partir d'une simplification de l'algorithme FTF (fast transversal filter) [29] [30], où le gain

d'adaptation est calculé en annulant complètement les prédicteurs aller et retour de l'algorithme FTF.

La mise à jour des coefficients du filtre adaptatif par l'algorithme FNLMS est donnée par:

$$\mathbf{w}(n) = \mathbf{w}(n-1) - \mu_{FNLMS} \gamma(n) \tilde{\mathbf{k}}(n) e(n) \quad (1.19)$$

Où $e(n)$ est l'erreur de filtrage a priori donnée par la relation (1.14), μ_{FNLMS} est une constante supplémentaire introduite pour mieux contrôler la stabilité de l'algorithme.

D'après [30], la relation d'adaptation du gain de Kalman dual $\tilde{\mathbf{k}}(n)$ peut s'écrire en faisant intervenir un prédicteur aller $\mathbf{a}(n)$ et prédicteur retour $\bar{\mathbf{r}}(n)$:

$$\begin{bmatrix} \tilde{\mathbf{k}}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}(n-1) \end{bmatrix} - \frac{\bar{e}_a(n)}{\lambda\alpha(n-1)} \begin{bmatrix} 1 \\ -\mathbf{a}(n-1) \end{bmatrix} + \frac{\bar{e}_r(n)}{\lambda\beta(n-1)} \begin{bmatrix} -\bar{\mathbf{r}}(n-1) \\ 1 \end{bmatrix} \quad (1.20)$$

Où $\bar{e}_a(n)$ et $\alpha(n)$ désignent respectivement l'erreur de prédiction aller et sa variance. Les variables $\bar{e}_r(n)$ et $\beta(n)$ sont l'erreur de prédiction retour et sa variance, respectivement. La constante λ est un facteur d'oubli exponentiel.

Dans l'algorithme FNLMS, le gain de Kalman dual $\tilde{\mathbf{k}}(n)$ est évalué en éliminant complètement les prédicteurs aller et retour ($\mathbf{a}(n) = \bar{\mathbf{r}}(n) = 0$) de (1.20) et en utilisant seulement une erreur de prédiction aller du signal d'entrée noté $\bar{e}(n)$ au lieu de $\bar{e}_a(n)$:

$$\begin{bmatrix} \tilde{\mathbf{k}}(n) \\ c(n) \end{bmatrix} = \begin{bmatrix} -\frac{\bar{e}(n)}{\lambda\alpha(n-1)+c_0} \\ \tilde{\mathbf{k}}(n-1) \end{bmatrix} \quad (1.21)$$

Où c_0 est une petite constante positive utilisée pour éviter les divisions par des petites valeurs en absence du signal d'entrée $x(n)$.

La variance de l'erreur de prédiction aller $\alpha(n)$ est donnée par:

$$\alpha(n) = \lambda\alpha(n-1) + \bar{e}^2(n) \quad (1.22)$$

L'utilisation de l'erreur de prédiction $\bar{e}(n)$ au lieu de $x(n)$ dans (1.21) est importante pour la vitesse de convergence car elle introduit un blanchiment du signal d'entrée $x(n)$ [28]. L'erreur de prédiction $\bar{e}(n)$ peut être calculée indépendamment du gain de Kalman dual par n'importe quel autre algorithme adaptatif. Afin d'obtenir un algorithme efficace en termes de complexité de calcul, l'erreur de prédiction $\bar{e}(n)$ est générée à l'aide d'un prédicteur du premier ordre:

$$\bar{e}(n) = x(n) - ax(n-1) \quad (1.23)$$

Où le coefficient de prédiction a est calculé en minimisant l'EQM $E(\bar{e}^2(n))$:

$$a = \frac{E[x(n)x(n-1)]}{E[x^2(n-1)]} = \frac{r_1}{r_0} \quad (1.24)$$

Où r_0 et r_1 désignent, respectivement, la puissance du signal d'entrée $x(n)$ et le premier coefficient de la fonction d'auto-corrélation du signal d'entrée $x(n)$. Le coefficient de prédiction a est évalué par des estimateurs récursifs comme suit:

$$a(n) = \frac{r_1(n)}{r_0(n) + c_a} \quad (1.25)$$

Avec:

$$r_1(n) = \lambda_a r_1(n-1) + x(n)x(n-1) \quad (1.26)$$

$$r_0(n) = \lambda_a r_0(n-1) + x^2(n) \quad (1.27)$$

Où, λ_a est un facteur d'oubli et c_a est une constante de régularisation.

Pour calculer la variable de vraisemblance $\gamma(n)$, plusieurs méthodes peuvent être utilisées. La première méthode utilise la définition directe :

$$\gamma(n) = \frac{1}{1 - \tilde{\mathbf{k}}^T(n)x(n)} \quad (1.28)$$

La complexité totale avec cette méthode est de l'ordre de $3L$ multiplications par itération.

La deuxième méthode exploite la propriété de décalage après la première composante $\tilde{k}^1(n)$ du gain de Kalman dual (1.21):

$$\gamma(n) = \frac{1}{1 - \sum_{i=1}^L \tilde{k}^1(n-i+1)x(n-i+1)} \quad (1.29)$$

Cette équation peut être réécrite comme suit:

$$\gamma(n) = \frac{1}{1 - \sum_{i=1}^L v(n-i+1)} \quad (1.30)$$

Où $v(n) = \tilde{k}^1(n)x(n)$ est le signal utilisé dans le processus de décalage.

La troisième méthode est obtenue en exploitant certaines propriétés d'invariance par décalage du vecteur signal d'entrée étendu à l'ordre $L + 1$:

$$\mathbf{x}^T(n) = [\mathbf{x}^T(n) \quad x(n-L)] \quad (1.31a)$$

$$\mathbf{x}^T(n) = [x(n) \quad \mathbf{x}^T(n-1)] \quad (1.31b)$$

En multipliant le premier élément et le second du côté gauche de l'expression (1.21) par (1.31a) et (1.31b) respectivement, on obtient:

$$\mathbf{x}^T(n)\tilde{\mathbf{k}}(n) + c(n)x(n-L) = \mathbf{x}^T(n-1)\tilde{\mathbf{k}}(n-1) - \frac{x(n)\bar{e}(n)}{\lambda\alpha(n-1)+c_0} \quad (1.32)$$

En substituant $\mathbf{x}^T(n)\tilde{\mathbf{k}}(n) = 1 - \gamma^{-1}(n)$ dans (1.32), on obtient:

$$\gamma^{-1}(n) = \gamma^{-1}(n-1) + c(n)x(n-L) + \frac{x(n)\bar{e}(n)}{\lambda\alpha(n-1)+c_0} \quad (1.33)$$

Cette dernière équation peut être écrite sous la forme récursive suivante:

$$\gamma(n) = \frac{\gamma(n-1)}{1+\gamma(n-1)\delta(n)} \quad (1.34)$$

Avec:

$$\delta(n) = c(n)x(n-L) + \frac{x(n)\bar{e}(n)}{\lambda\alpha(n-1)+c_0} \quad (1.35)$$

- *Complexité de calcul*

En utilisant l'équation (1.30) ou (1.34) pour calculer la variable de vraisemblance $\gamma(n)$, la complexité de calcul de l'algorithme FNLMS est de l'ordre de $2L$ multiplications par itération.

L'avantage de l'algorithme FNLMS est qu'il présente une complexité de calcul réduite similaire à celle de l'algorithme NLMS avec de meilleures performances en vitesse de convergence et en capacité de poursuite [28].

Les équations de fonctionnement de l'algorithme FNLMS avec leurs complexités de calcul sont listées dans le tableau 1.3.

Tableau 1.3 :Complexité de calcul de l'algorithme FNLMS.

Etapes de l'algorithme	Nombre de Mult/Div
Etape de Prédiction:	
Erreur de prédiction:	
$\bar{e}(n) = x(n) - ax(n - 1)$	1
$a(n) = \frac{r_1(n)}{r_0(n) + c_a}$	1
$r_1(n) = \lambda_a r_1(n - 1) + x(n)x(n - 1)$	2
$r_0(n) = \lambda_a r_0(n - 1) + x^2(n)$	2
Variance de l'erreur de prédiction aller:	
$\alpha(n) = \lambda\alpha(n - 1) + \bar{e}^2(n)$	2
Gain de Kalman dual:	
$\begin{bmatrix} \tilde{\mathbf{k}}(n) \\ c(n) \end{bmatrix} = \begin{bmatrix} -\frac{\bar{e}(n)}{\lambda\alpha(n-1)+c_0} \\ \tilde{\mathbf{k}}(n-1) \end{bmatrix}$	2
Variable de vraisemblance:	
Version 1:	
$\delta(n) = c(n)x(n - L) + \frac{x(n)\bar{e}(n)}{\lambda\alpha(n - 1) + c_0}$	4
$\gamma(n) = \frac{\gamma(n - 1)}{1 + \gamma(n - 1)\delta(n)}$	2
Version 2:	
$v(n) = \tilde{k}^1(n)x(n)$	
$\gamma(n) = \frac{1}{1 - \sum_{i=1}^L v(n - i + 1)}$	
Etape de filtrage:	
$e(n) = d(n) - \mathbf{w}^T(n - 1)\mathbf{x}(n)$	L
Etape d'adaptation:	
$\mathbf{w}(n) = \mathbf{w}(n - 1) - \mu_{FNLMS}\tilde{\mathbf{k}}(n)\gamma(n)e(n)$	$L + 2$
Complexité de calcul Totale	$2L + 18$

1.5. Critères de comparaison et choix des algorithmes adaptatifs

Dans la comparaison des performances entre les algorithmes de filtrage adaptatifs, plusieurs critères de performances peuvent être utilisés pour évaluer objectivement les performances des nouvelles versions algorithmiques proposées dans la littérature et celle déjà publiées. Ces critères sont résumés comme suit [31]:

- Taux de convergence.
- Capacité de poursuite.
- Robustesse au bruit.
- Complexité de calcul.
- Stabilité numérique

1.6. Conclusion

Dans ce chapitre, des notions théoriques ont été présentées. Ces notions couvrent le filtrage optimal de Wiener, le filtrage adaptatif et ses applications. Les principaux algorithmes du filtrage adaptatif tels que le LMS, NLMS et FNLMS ont été ainsi présentés.

Les algorithmes du gradient stochastique (LMS et NLMS) sont caractérisés par leur complexité de calcul réduite ainsi que leur stabilité numérique, Cependant ils souffrent d'une convergence lente avec les signaux corrélés.

L'algorithme NLMS à convergence rapide obtenu à partir d'une simplification de l'algorithme FTF est un bon compromis entre ces deux algorithmes (bonne performance de vitesse de convergence et complexité de calcul réduite).

CHAPITRE 2

GENERALITES SUR LES METHODES DE REDUCTION DU BRUIT ACOUSTIQUE ET LE REHAUSSEMENT DU SIGNAL DE PAROLE.

2.1. Introduction

Dans ce chapitre nous présenterons un aperçu des méthodes de réduction du bruit acoustique. La classification adoptée pour ces méthodes repose sur le nombre de microphones disponibles pour le traitement.

Nous traiterons en premier, le cas des méthodes de débruitage mono-capteur, où un seul microphone est utilisé. Deux techniques standards de réduction du bruit mono-capteur seront exposées dans la partie 2.2.1.

Nous aborderons ensuite, le cas où l'on dispose soit d'un réseau de microphones soit de deux microphones seulement. Pour cela nous présenterons dans la partie 2.2.2 les méthodes multi-capteurs les plus utilisées dans le domaine de débruitage de la parole.

Finalement, nous formulerons le problème de la séparation aveugle de sources dans un contexte général, puis nous nous intéresserons au problème de réduction du bruit acoustique par les techniques de séparation aveugle de sources bicapteurs.

2.2. Réduction du bruit acoustique

Le domaine de débruitage est un domaine de recherche très actif et présent dans de nombreux champs d'applications du traitement du signal. L'objectif des techniques de réduction du bruit est l'amélioration de la qualité et l'intelligibilité du signal vocal et par conséquent l'amélioration des performances des applications en relation. La classification de ces techniques repose sur le nombre de capteurs disponible pour le traitement, où, on peut distinguer deux catégories de base, à savoir: les méthodes de réduction du bruit mono-capteur et multi-capteurs [32]. La figure 2.1 illustre les deux catégories de base des méthodes de réduction du bruit, qui seront décrites par la suite.

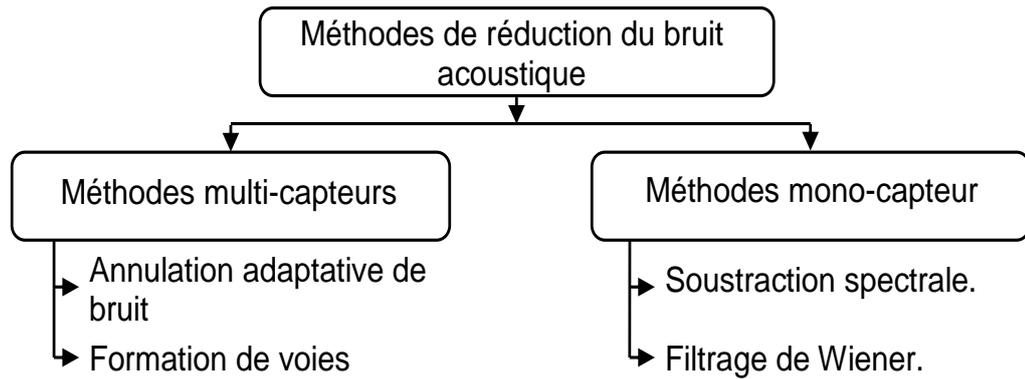


Figure 2. 1 : Catégorisation des méthodes de réduction du bruit.

3.3.1. Méthodes de débruitage mono-capteur

Les méthodes de débruitage mono-capteur utilisent un seul microphone pour capter le signal de parole dégradé par le bruit. Les traitements associés utilisent donc un seul signal pour assurer le débruitage. Nous présenterons dans cette partie, les méthodes qui sont à base du traitement mono-capteur dont, le modèle de débruitage utilisé pour ces méthodes est illustré sur la figure 2.2.

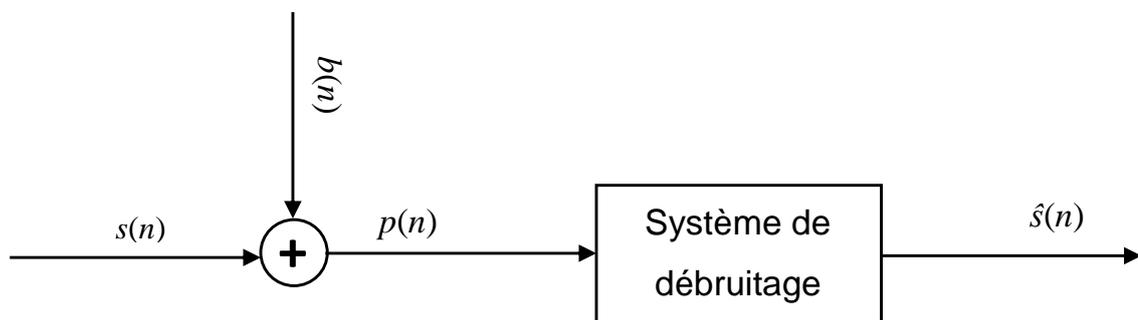


Figure 2. 2 : Modèle de débruitage mono-capteur.

A partir de ce modèle, on aura:

$$p(n) = s(n) + b(n) \quad (2.1)$$

Où, $p(n)$, $s(n)$ et $b(n)$ désignent respectivement le signal observé bruité, le signal de parole propre et le bruit additif.

La plupart des techniques de débruitage mono-capteur réalisent l'estimation du signal de parole propre $s(n)$ à partir du signal observé $p(n)$ dans le domaine fréquentiel. Le signal de parole peut être vu comme une réalisation particulière d'un processus aléatoire stationnaire sur des intervalles de temps de courte durée. Ses caractéristiques statistiques peuvent, donc, être estimées sur des tranches

temporelles de durée (20 à 40ms). C'est pour cette raison que l'on considère les spectres instantanés, obtenus à l'aide de la transformée de Fourier à court terme sur des termes successifs [33].

Dans le domaine fréquentiel, la relation (2.1) s'écrit:

$$P(f, k) = S(f, k) + B(f, k) \quad (2.2)$$

Où, $P(f, k)$, $S(f, k)$ et $B(f, k)$ désignent respectivement les transformées de Fourier à court terme du signal observé, du signal de parole propre et du bruit additif, et f désigne l'indice fréquentiel et k l'indice de la trame courante. Cette dernière relation est justifiée sous l'hypothèse de la non-corrélation entre le signal de parole et du bruit.

- *Soustraction spectrale*

La soustraction spectrale est la méthode de débruitage la plus ancienne et sans doute la plus facile à mettre en œuvre. Elle a été introduite par Boll [34]. Elle est basée sur l'estimation à priori du bruit supposé stationnaire, qui peut être apprise lors des phases de silences. Il existe deux versions de base pour la soustraction spectrale se différenciant par l'amplitude ou la puissance.

Soustraction spectrale d'amplitude (SAA): cette technique est décrite par l'équation suivante [35]:

$$|\hat{S}(f, k)| = |P(f, k)| - |\hat{B}(f, k)| \quad (2.3)$$

Où $|\hat{S}(f)|$ est l'estimé du spectre d'amplitude du signal de parole propre et $|\hat{B}(f)|$ est l'estimé du spectre d'amplitude du bruit.

Cette dernière relation n'a pas de sens physique que si $|\hat{S}(f)|$ est positif. Dans le cas où l'équation (2.3) donne un résultat négatif, On peut le rendre positif en changeant de signe ou bien en l'annulant comme dans l'équation (2.4):

$$|\hat{S}(f, k)| = \begin{cases} |P(f, k)| - |\hat{B}(f, k)| & \text{si } |P(f, k)| \geq |\hat{B}(f, k)| \\ 0 & \text{sinon} \end{cases} \quad (2.4)$$

Soustraction spectrale en puissance (SSP): la SSP est basée sur l'estimation de la densité spectrale de puissance du bruit, et elle est décrite par la relation suivante [35]:

$$|\hat{S}(f, k)|^2 = |P(f, k)|^2 - |\hat{B}(f, k)|^2 \quad (2.5)$$

Où, $|\hat{S}(f)|^2$ est l'estimé du spectre de puissance du signal de parole propre et $|\hat{B}(f)|^2$ est l'estimé du spectre de puissance du bruit.

De la même façon que pour la SSA, les valeurs négatives du module carré $|\hat{S}(f, k)|^2$ sont à proscrire ce qui amène à contraindre le résultat :

$$|\hat{S}(f, k)|^2 = \begin{cases} |P(f, k)|^2 - |\hat{B}(f, k)|^2 & \text{si } |P(f, k)|^2 \geq |\hat{B}(f, k)|^2 \\ 0 & \text{sinon} \end{cases} \quad (2.6)$$

Afin de restituer le signal débruité dans le domaine temporel, on effectue une transformée de Fourier discrète inverse qui nécessite la connaissance de la phase du signal de parole propre. Comme il n'y a pas de méthodes aisées pour estimer la phase du signal propre, la valeur de la phase du signal observé est utilisée.

$$\hat{s}(n) = IFFT[|\hat{S}(f, k)|e^{j\theta_p(f, k)}] \quad (2.7)$$

- *Filtrage de Wiener*

Cette technique de débruitage consiste à obtenir une estimation du signal de parole propre d'une façon optimale, au sens du minimum de l'erreur quadratique moyenne, dans le domaine fréquentiel [36].

L'erreur d'estimation $\epsilon(f, k)$ est définie comme la différence entre les transformées de Fourier à court terme du signal de parole propre $S(f, k)$ et du signal filtré $\hat{S}(f, k)$.

$$\begin{aligned} E[\epsilon(f, k)^2] &= E[|S(f, k) - \hat{S}(f, k)|^2] \\ &= E[|S(f, k) - W(f, k)P(f, k)|^2] \end{aligned} \quad (2.8)$$

En tenant compte de la non corrélation des signaux $s(n)$ et $b(n)$ pour chaque trame, et en annulant le gradient de $E[\epsilon(f, k)^2]$ par rapport au filtre $W(f, k)$, on peut calculer le filtre de Wiener $W(f, k)$ comme suit:

$$W(f, k) = \frac{|S(f, k)|^2}{|S(f, k)|^2 + |B(f, k)|^2} \quad (2.9)$$

En utilisant la définition du RSB à priori, le filtre de Wiener peut s'exprimer sous cette forme [37]:

$$W(f, k) = \frac{RSB_{prio}(f, k)}{1 + RSB_{prio}(f, k)} \quad (2.10)$$

Avec: $RSB_{prio} = \frac{|S(f, k)|^2}{|B(f, k)|^2}$.

3.3.2. Méthodes de débruitage Multi-capteurs

Les méthodes de réduction du bruit mono-capteur, précédemment décrites, se fondent seulement sur les informations temporelles ou spectrales contenues dans le signal d'entrée. Ils peuvent être efficaces sous l'hypothèse de stationnarité du bruit, mais ils rapportent généralement une déformation du signal désiré, en revanche avec les méthodes de réduction du bruit multi-capteurs, le bruit peut être réduit sans l'affectation de l'intelligibilité du signal de parole [38].

Considérant le système de débruitage multi-capteurs représenté par la figure 2.3. Supposons que l'on dispose d'un réseau de C capteurs recevant la contribution de deux signaux sources. La première source est le signal de parole $s(n)$, tandis que la seconde source est le signal de bruit $b(n)$. Les signaux observés à la sortie des capteurs peuvent être modélisés comme la somme des convolutions entre les signaux sources et les réponses impulsionnelles des différents chemins (direct et indirect) de propagation entre les sources et les capteurs [39].

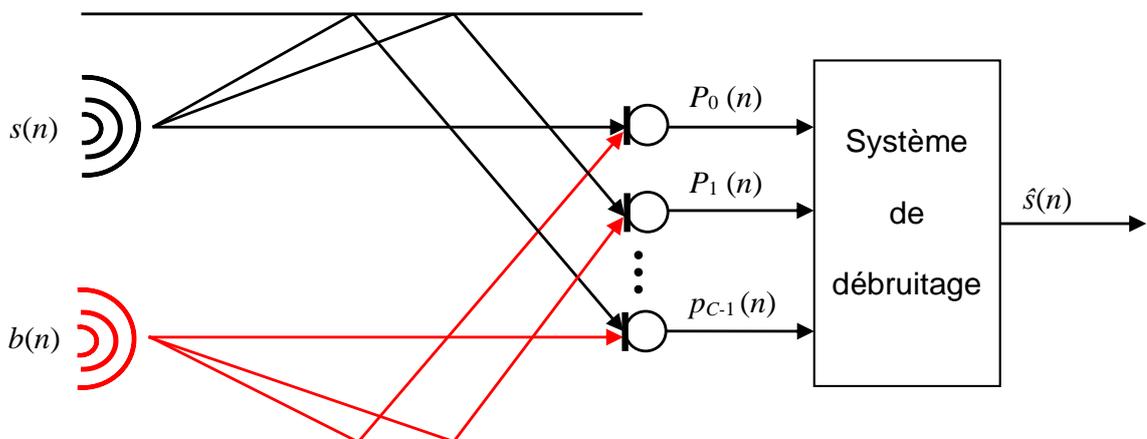


Figure 2. 3 : Modèle de débruitage multi-capteurs.

Le signal observé $p_j(n)$ au $j^{\text{ième}}$ capteur peut être modélisé comme suit [39]:

$$p_j(n) = \sum_{m=0}^{L-1} s(m)h_{sj}(n-m) + \sum_{m=0}^{L-1} b(m)h_{bj}(n-m) \quad j = 0, 1, \dots, C-1 \quad (2.11)$$

Où, h_{sj} représente la réponse du canal acoustique entre la source de parole et le $j^{\text{ième}}$ capteur et h_{bj} est la réponse du canal acoustique entre la source de bruit et le $j^{\text{ième}}$ capteur, modélisées par des filtres à réponse impulsionnelle finie.

Afin d'estimer le signal de parole en sortie, plusieurs techniques de débruitage multi-capteurs ont été proposées dans la littérature [40] [41]. Nous

présenterons dans cette partie en premier lieu, la méthode la plus connue et la plus ancienne qui est l'annulation adaptative du bruit (ANC : adaptive noise cancellation), puis nous présenterons brièvement les méthodes de débruitage multi-capteurs basées sur la combinaison des filtrages temporel et spatial par les approches de traitement d'antenne linéaire.

- *Annulation adaptative du bruit (ANC)*

L'annulation adaptative du bruit est une technique alternative d'estimation des signaux corrompus par du bruit additif, où deux microphones sont utilisés. Le premier microphone capte le signal de parole bruité, tandis que le second microphone capte uniquement le bruit (référence de bruit). Sous certaines hypothèses, il est possible d'annuler le bruit sur le premier microphone par filtrage adaptatif, à partir de la référence de bruit [42], comme c'est illustré dans la figure 2.4.

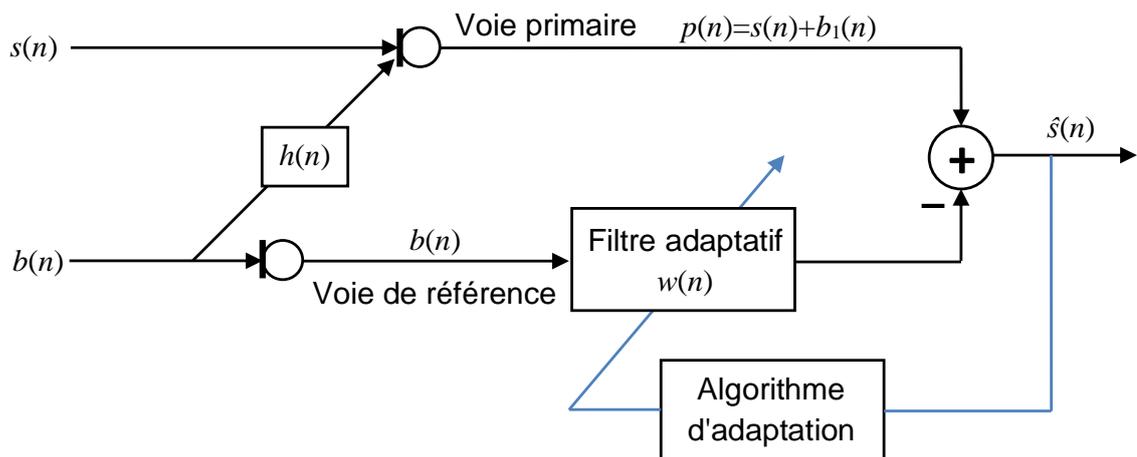


Figure 2. 4 : Structure d'annulation adaptative du bruit.

A partir de la figure 2.4, Le signal observé bruité $p(n)$ sur la voie primaire du système ANC est donné par:

$$p(n) = s(n) + b_1(n) \quad (2.12)$$

Où $b_1(n)$ est un bruit additif décorrélé du signal de parole propre $s(n)$.

La voie de référence reçoit uniquement le bruit $b(n)$ qui est fortement corrélé à $b_1(n)$ mais décorrélé de $s(n)$. La relation entre $b(n)$ et $b_1(n)$ est donnée par:

$$b_1(n) = b(n) * h(n) \quad (2.13)$$

Où $(*)$ représente le produit de convolution et $h(n)$ représente la réponse impulsionnelle.

L'objectif de la technique d'annulation adaptative du bruit est de filtrer la référence $b(n)$ de manière adéquate afin d'avoir une estimation du bruit $b_1(n)$, qui sera par la suite soustraite du signal bruité $p(n)$, ce qui produit en sortie une estimation $\hat{s}(n)$ du signal de parole propre $s(n)$.

En tenant compte de la relation linéaire entre les bruits dans les deux voies, le signal de sortie $\hat{s}(n)$ peut s'écrire sous la forme:

$$\hat{s}(n) = s(n) - (h(n) - w(n)) * b(n) \quad (2.14)$$

La solution optimale est obtenue lorsque $h(n) = w^{opt}(n)$.

Une solution pour obtenir $w(n)$ consiste à effectuer une estimation adaptative de la réponse impulsionnelle $h(n)$, on choisit alors une modélisation paramétrique de $w(n)$ sous forme d'un filtre à réponse impulsionnelle finie dont on estime les coefficients de $w(n)$ suivant un algorithme adaptatif. Pour la mise à jour des coefficients du filtre adaptatif $w(n)$, on peut utiliser plusieurs algorithmes adaptatifs comme les algorithmes de type gradient stochastique (LMS et NLMS) [43].

- *Formation de voies*

Formation de voies ou Beamforming en anglais est l'une des plus importantes techniques de traitement d'antenne linéaire. Elle peut être considérée comme un filtre spatial qui opère sur chaque sortie du réseau de microphones. Ce filtre va permettre de récupérer des signaux dans des directions particulières avec un gain maximum, i.e. de favoriser le gain pour les directions d'intérêt et d'atténuer les signaux dans les directions non souhaitées (interférences) [38]. On peut différencier deux catégories de formation de voies: formation de voies fixes [44] et formation de voies adaptatives [45].

2.3. Réduction du bruit acoustique par la séparation aveugle de sources

Le problème de la séparation aveugle de sources (en anglais: blind source separation BSS) a été formulé pour la première fois au début des années 80 par Christian Jutten, Jeanny Hérault et Bernard Ans dans le cadre de la modélisation neuronale [46]. Plus tard, le problème de la BSS a été largement abordé dans plusieurs domaines d'applications tels que: radiocommunication, rehaussement de la parole, traitement d'image, génie biomédical, radar, etc.

La séparation aveugle de sources consiste à récupérer un ensemble de signaux sources inconnus à partir d'un ensemble de signaux observés qui résultent d'un mélange inconnu de ces signaux sources. Le terme aveugle se réfère au fait que ni les signaux sources, ni les mélanges sont connus [47]. La plupart des techniques de BSS reposent sur l'hypothèse de l'indépendance des sources.

Nous présenterons dans cette partie une formulation mathématique du problème de la BSS dans un contexte général, ainsi que les différents types de mélanges, les hypothèses classiques faites souvent sur les sources, bruits et le système de mélange. Enfin, nous aborderons le problème de réduction du bruit acoustique par les méthodes de séparation aveugle de sources bicapteurs.

2.3.1. Modélisation du problème

Dans un contexte général, la séparation aveugle de sources vise à estimer un ensemble de N signaux sources $s_i(n)$, avec $1 \leq i \leq N$, à partir d'un ensemble de C signaux observés $p_j(n)$, avec $1 \leq j \leq C$, qui sont des mélanges de ces signaux sources. La relation liant les sources et les observations est donnée par:

$$\mathbf{p}(n) = \mathbf{H}[\mathbf{s}(n)] + \boldsymbol{\eta}(n) \quad (2.15)$$

Où $\mathbf{s}(n) = [s_1(n), \dots, s_N(n)]^T$ et $\mathbf{p}(n) = [p_1(n), \dots, p_C(n)]^T$ sont deux vecteurs colonnes représentant respectivement l'ensemble des signaux sources et l'ensemble des signaux d'observations. $\boldsymbol{\eta}(n) = [\eta_1(n), \dots, \eta_C(n)]^T$ est le vecteur de bruit qui modélise les erreurs de mesure des capteurs, et $\mathbf{H}(\cdot)$ dénote l'opérateur de mélange, $(\cdot)^T$ désigne la transposée et n l'indice de temps.

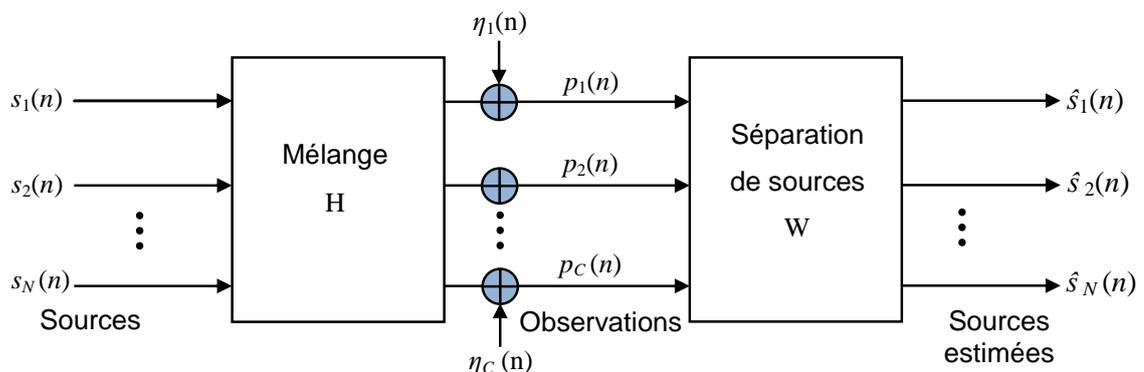


Figure 2. 5 : Structure générale de la séparation aveugle de sources.

L'objectif principal de la séparation aveugle de sources est de construire un système de séparation W , qui permet d'estimer au mieux les signaux sources. La

reconstruction des signaux sources en absence de bruit $\eta(n)$, est réalisée grâce au modèle [48]:

$$\hat{\mathbf{s}}(n) = \mathbf{W}[\mathbf{p}(n)] \quad (2.16)$$

Où $\hat{\mathbf{s}}(n) = [\hat{s}_1(n), \dots, \hat{s}_N(n)]^T$ est le vecteur des sources estimées. La structure générale de la séparation aveugle de sources est illustrée dans la figure 2.5.

Les méthodes de séparation aveugle de sources peuvent être classées principalement selon trois critères : le type de mélange, le nombre de signaux observés par rapport au nombre de signaux sources, et les propriétés des sources considérées. Ces différents points seront détaillés dans les paragraphes suivants.

2.3.2. Types de mélanges

Le mélange peut être linéaire ou non linéaire, instantané ou convolutif. Aussi, il peut être variant ou invariant dans le temps. En général, c'est le modèle linéaire invariant dans le temps qui est considéré dans la plupart des applications [49]. Nous nous intéresserons dans ce paragraphe seulement au modèle linéaire invariant dans le temps où l'on considère plus particulièrement les cas des mélanges instantanés et convolutifs.

- *Mélange linéaire instantané*

Tout d'abord, nous commencerons par décrire le modèle de mélange linéaire instantané. Dans ce cas, l'opérateur de mélange H correspond à une somme pondérée et instantanée des signaux sources [48]. Le modèle liant la $j^{\text{ième}}$ observation aux N sources à un instant n est donné par:

$$p_j(n) = \sum_{i=1}^N h_{ij} s_i(n) + \eta_j(n); \quad j = 1, \dots, C \quad (2.17)$$

Où, h_{ij} sont des coefficients scalaires du mélange linéaire instantané.

- *Mélange linéaire convolutif*

Le mélange convolutif est un système de filtres dont les réponses impulsionnelles permettent de modéliser les propagations le long de trajets multiples entre sources et observations. Dans ce cas, l'opérateur de mélange H représente un filtre linéaire et invariant dans le temps [50]. Ce modèle se traduit par l'équation suivante:

$$p_j(n) = \sum_{i=1}^N h_{ij}(n) * s_i(n) + \eta_j(n); \quad j = 1, \dots, C \quad (2.18)$$

Où, $*$ est le produit de convolution, $h_{ij}(n)$ représente la réponse impulsionnelle entre la $i^{\text{ième}}$ source et le $j^{\text{ième}}$ capteur.

2.3.3. Détermination des mélanges

Nous définissons la détermination du mélange Δ comme la différence entre le nombre de sources et d'observations [50]. Trois types de mélanges se déclinent à partir de cette définition, à savoir:

- Si $\Delta < 0$, alors il y a plus d'observations que de sources ($C > N$), nous parlons de mélange sur-déterminé.
- Si $\Delta = 0$, alors il y a autant de sources que d'observations ($C = N$), nous parlons de mélange déterminé.
- Si $\Delta > 0$, alors il y a plus de sources que d'observations ($C < N$), nous parlons de mélange sous-déterminé.

2.3.4. Hypothèses classiques de BSS

La séparation est dite aveugle si on est capable de séparer les signaux observés sans aucune information sur le système de mélange et que les sources sont inobservables. Sans hypothèses supplémentaires la BSS apparaît comme un problème insoluble. Nous résumerons dans ce paragraphe les hypothèses sur lesquelles la BSS est basée [50].

- Hypothèse 1: les signaux sources $s_i(n)$ sont statistiquement indépendants.
- Hypothèse 2: les signaux sources $s_i(n)$ sont des processus aléatoires stationnaires.
- Hypothèse 3: le nombre des observations est supérieur ou égal au nombre de sources ($C \geq N$).
- Hypothèse 4: le bruit $\eta(n)$ est additif, et statistiquement indépendant des signaux sources.

2.3.5. Séparation aveugle de sources bicapteurs

Dans le cadre de cette thèse, nous allons traiter le problème de la réduction du bruit acoustique et le rehaussement du signal de parole par les techniques de séparation aveugle de sources bicapteurs, où l'on considère le cas de mélange convolutif linéaire déterminé d'ordre 2 ($C = N = 2$).

▪ *Mélange linéaire convolutif bicapteurs*

Considérons une situation de prise de son par deux microphones, une source de parole se situe près du 1^{er} microphone, l'autre source de perturbation (bruit) est près du 2^{ème} microphone. A la sortie des microphones, on observe une

superposition des signaux primitifs selon un mélange convolutif [51]. Le modèle général du mélange convolutif bicapteurs est illustré sur la figure 2.6.

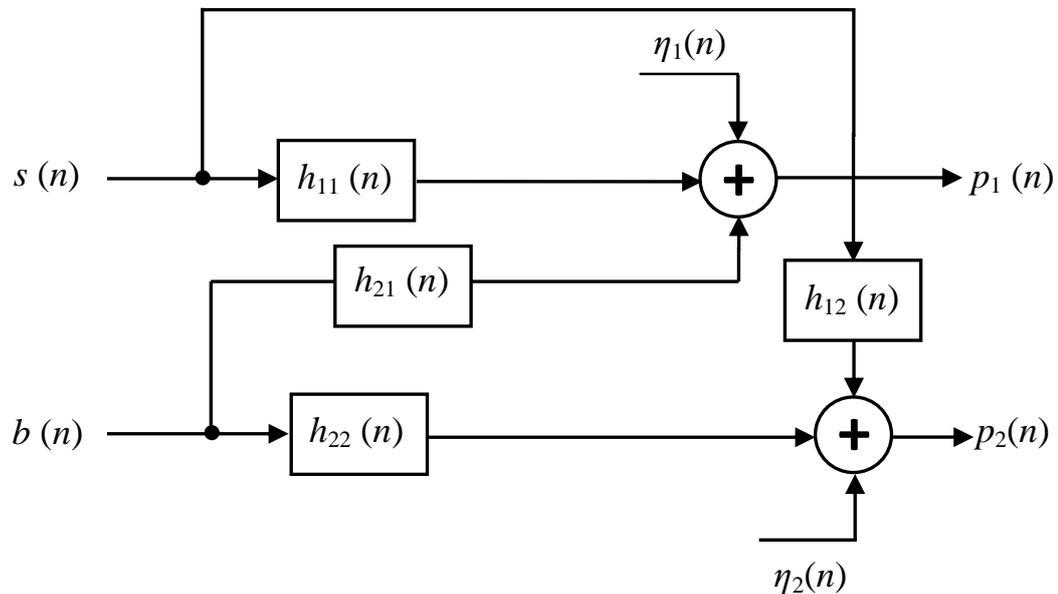


Figure 2. 6 : Modèle général du mélange convolutif.

A partir de la figure 2.6, Les deux signaux observés $p_1(n)$ et $p_2(n)$ à la sortie des microphones sont donnés par:

$$p_1(n) = h_{11}(n) * s(n) + h_{21}(n) * b(n) + \eta_1(n) \quad (2.19)$$

$$p_2(n) = h_{22}(n) * b(n) + h_{12}(n) * s(n) + \eta_2(n) \quad (2.20)$$

Où les réponses impulsionnelles $h_{11}(n)$ et $h_{22}(n)$ représentent le couplage acoustique direct entre les sources et les microphones. Les réponses impulsionnelles $h_{12}(n)$, $h_{21}(n)$ représentent le couplage acoustique croisé entre les sources et les microphones. $\eta_1(n)$ et $\eta_2(n)$ sont les erreurs de mesure.

▪ *Mélange convolutif bicapteurs simplifié*

Pour simplifier le modèle du mélange convolutif qui est présenté dans la figure (2.6), certaines suppositions sont prises en compte telles que [51]:

- Les sources sont ponctuelles.
- Un microphone est placé près du locuteur et l'autre près de la source de perturbation.
- Les canaux directs $h_{11}(n)$ et $h_{22}(n)$ sont égaux à l'impulsion de Kroneker $\delta(n)$.
- Les erreurs de mesure des microphones $\eta_1(n)$ et $\eta_2(n)$ sont négligeables.
- La distance entre les microphones est petite.

Dans ces conditions un modèle simplifié du mélange convolutif est présenté dans la figure 2.7.

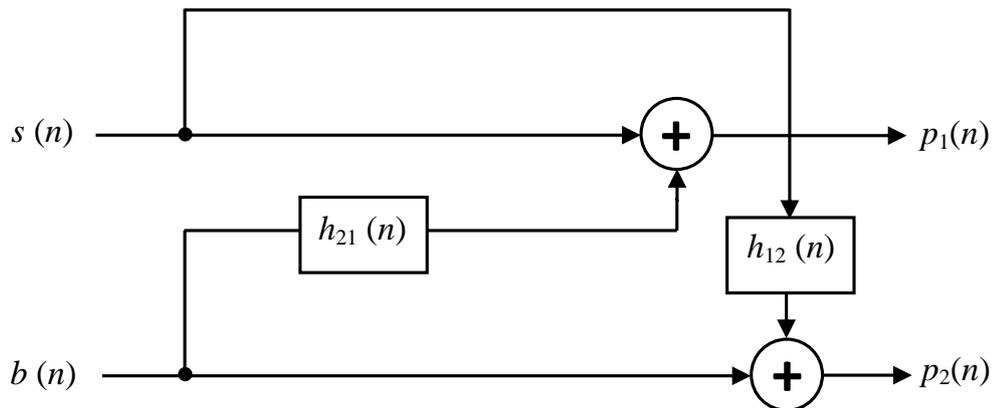


Figure 2. 7 : Modèle simplifié du mélange convolutif.

A partir de la figure 2.7, Les deux signaux observés $p_1(n)$ et $p_2(n)$ à la sortie des microphones sont donnés par:

$$p_1(n) = s(n) + h_{21}(n) * b(n) \quad (2.21)$$

$$p_2(n) = b(n) + h_{12}(n) * s(n) \quad (2.22)$$

- *Modèle de séparation*

Le problème de la séparation aveugle de sources consiste à retrouver, avec le moins de connaissance à priori, les signaux utiles qui ont été mélangés. Pour résoudre ce problème deux structures appelées directe et récursive, sont fréquemment utilisées. Ces deux structures ont été largement exploitées dans le domaine de réduction du bruit acoustique et le rehaussement du signal de parole.

Structure directe: la structure de séparation aveugle de sources directe (en anglais: forward blind source separation (FBSS)) a pour but d'estimer les deux signaux sources $s(n)$ et $b(n)$, à partir seulement des signaux observés $p_1(n)$ et $p_2(n)$. Elle est basée principalement sur l'hypothèse de l'indépendance des signaux sources, en outre cette structure à besoin de deux filtres adaptatifs $w_{21}(n)$ et $w_{12}(n)$ pour récupérer les signaux d'origine [52]. La structure de séparation aveugle de sources directe est illustrée sur la figure 2.8.

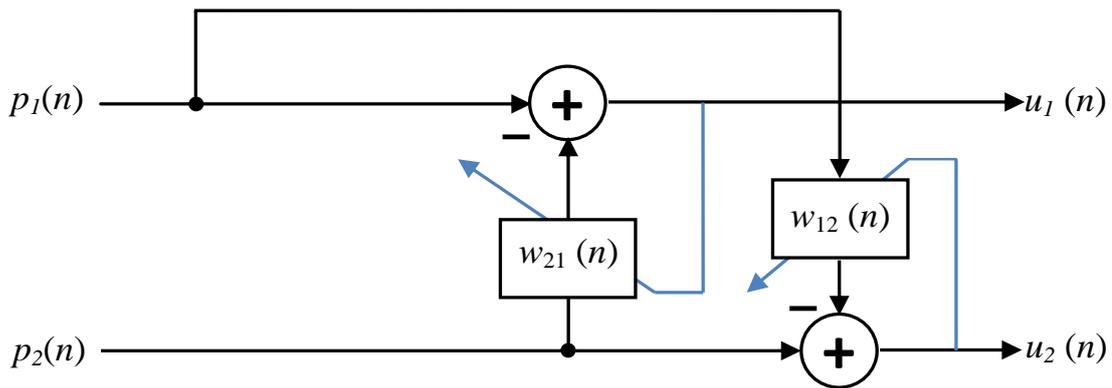


Figure 2. 8 : Structure de séparation aveugle de sources directe (FBSS).

Les deux signaux de sortie de la structure directe sont donnés par les formules suivantes:

$$u_1(n) = p_1(n) - p_2(n) * w_{21}(n) \quad (2.23)$$

$$u_2(n) = p_2(n) - p_1(n) * w_{12}(n) \quad (2.24)$$

En remplaçant les équations (2.21) et (2.22) dans (2.23) et (2.24), nous obtiendrons:

$$u_1(n) = b(n) * [h_{21}(n) - w_{21}(n)] + s(n) * [\delta(n) - h_{12}(n) * w_{21}(n)] \quad (2.25)$$

$$u_2(n) = s(n) * [h_{12}(n) - w_{12}(n)] + b(n) * [\delta(n) - h_{21}(n) * w_{12}(n)] \quad (2.26)$$

Pour l'obtention du signal de parole sur la sortie $u_1(n)$ et le bruit sur la sortie $u_2(n)$, une hypothèse d'optimalité est utilisée pour les deux filtres adaptatifs ($w_{21}^{opt}(n) = h_{21}(n)$ et $w_{12}^{opt}(n) = h_{12}(n)$), dans ce cas les sorties $u_1(n)$ et $u_2(n)$ deviennent:

$$u_1(n) = s(n) * [\delta(n) - h_{12}(n) * w_{21}(n)] \quad (2.27)$$

$$u_2(n) = b(n) * [\delta(n) - h_{21}(n) * w_{12}(n)] \quad (2.28)$$

A partir des relations (2.27) et (2.28), nous remarquerons que les sorties $u_1(n)$ et $u_2(n)$ convergent respectivement vers les deux signaux originaux $s(n)$ et $b(n)$ avec une petite distorsion. La correction de cette distorsion nécessite des post-filtres ($pf_1 = pf_2 = \frac{1}{\delta(n) - w_{12}(n) * w_{21}(n)}$) placés en sortie de la structure directe. Plusieurs techniques adaptatives ont été proposées dans la littérature afin d'estimer ces post-filtres [53]. Dans le reste de ce document, nous considérons la situation où les deux microphones sont espacés, ce qui nous permet d'éviter le problème des post-filtres.

Structure récurrente: La forme classique de la structure de séparation aveugle de sources récurrente (en anglais: backward blind source separation (BBSS)) est illustrée sur la figure 2.9. Dans cette structure, la sortie $v_i(n)$ du système de séparation est une combinaison linéaire d'une observation $p_i(n)$ et de l'autre sortie $v_j(n)$, avec $i \neq j \in \{1, 2\}$, i.e. afin d'estimer le signal de sortie $v_1(n)$, on utilise le deuxième signal de sortie $v_2(n)$, et vice-versa). Cette structure est basée principalement sur l'hypothèse de l'indépendance des signaux sources [52].

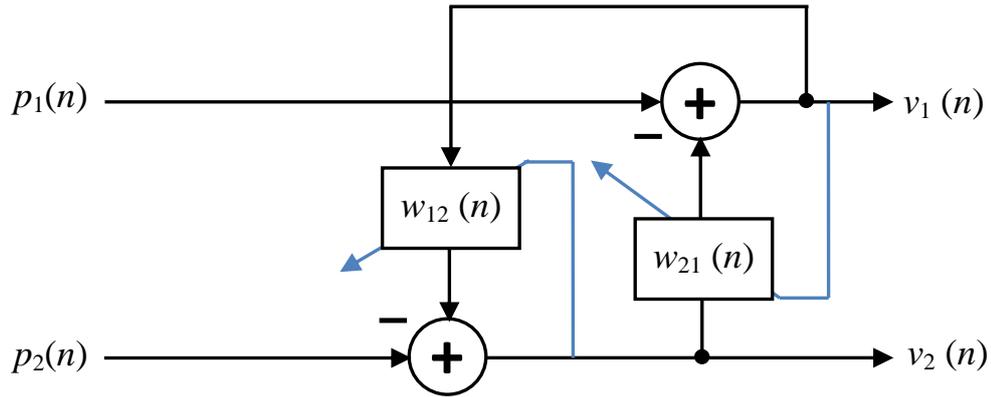


Figure 2. 9 : Structure de séparation aveugle de sources récurrente (BBSS).

Les deux signaux de sortie $v_1(n)$ et $v_2(n)$ de la structure récurrente sont donnés par les formules suivantes:

$$v_1(n) = p_1(n) - v_2(n) * w_{21}(n) \quad (2.29)$$

$$v_2(n) = p_2(n) - v_1(n) * w_{12}(n) \quad (2.30)$$

En remplaçant les équations (2.21) et (2.22) dans (2.29) et (2.30), nous obtiendrons :

$$v_1(n) = \left(\frac{1}{\delta(n) - w_{21}(n) * w_{12}(n)} \right) * [b(n) * (h_{21}(n) - w_{21}(n)) + s(n) * (\delta(n) - h_{12}(n) * w_{21}(n))] \quad (2.31)$$

$$v_2(n) = \left(\frac{1}{\delta(n) - w_{21}(n) * w_{12}(n)} \right) * [s(n) * (h_{12}(n) - w_{12}(n)) + b(n) * (\delta(n) - h_{21}(n) * w_{12}(n))] \quad (2.32)$$

La solution optimale est obtenue lorsque $w_{21}^{opt}(n) = h_{21}(n)$ et $w_{12}^{opt}(n) = h_{12}(n)$, alors les sorties $v_1(n)$ et $v_2(n)$ deviennent:

$$v_1(n) = s(n) \quad (2.33)$$

$$v_2(n) = b(n) \quad (2.34)$$

A partir des relations (2.33) et (2.34), nous remarquerons que les sorties $v_1(n)$ et $v_2(n)$ convergent respectivement vers les deux signaux originaux $s(n)$ et $b(n)$ sans aucune distorsion. C'est l'avantage majeur de cette structure par rapport à la FBSS qui nécessite des post-filtres à la sortie pour corriger la distorsion.

2.4. Conclusion

Dans ce chapitre, nous avons présenté un état de l'art des méthodes de débruitage de la parole les plus communément utilisées.

Les méthodes de débruitage mono-capteur présentées dans ce chapitre, comme la soustraction spectrale et le filtrage de Wiener, réussissent à séparer les composantes de bruit du signal de parole en s'appuyant sur des techniques d'estimation de bruit, mais en contrepartie, introduisent un bruit résiduel gênant pour la perception humaine. De plus, ces méthodes requièrent une connaissance a priori sur les signaux de bruit et de parole, et supposent que le bruit est stationnaire. Par contre, la technique d'annulation adaptative de bruit (ANC), qui rentre dans la catégorie des méthodes de débruitage multi-capteurs décrite dans ce chapitre, ne nécessite aucune connaissance a priori des statistiques de bruit ou de la parole, et elle est adéquate pour le cas des bruits non stationnaires. Cependant les performances de cette méthode sont limitées dans les applications réelles. Ceci est dû principalement à la difficulté de disposer d'un signal de référence de bruit seul.

Dans le cas des méthodes de débruitage multi-capteurs, nous avons ainsi présenté brièvement la technique de formation de voies. Cette méthode utilise un réseau de microphones afin d'ajouter des informations spatiales aidant le débruitage de la parole.

Enfin, nous avons traité le problème de débruitage de la parole par les techniques de séparation aveugle de sources bicapteurs (structure directe, et récursive).

CHAPITRE 3

ALGORITHMES BICAPTEURS POUR LE REHAUSSEMENT DU SIGNAL DE PAROLE

3.1. Introduction

Dans les nouveaux systèmes de télécommunications sans fil, tels que la téléphonie mains-libres, la qualité et l'intelligibilité du signal vocal transmis est souvent dégradée par la présence du bruit ambiant.

La gêne due à la perturbation qualifiée de bruit est généralement source d'inconfort et de fatigue pour le locuteur distant, ce qui rend la conversation téléphonique rapidement fastidieuse. Dans ces conditions, il se ressent le besoin d'un traitement capable de réduire la gêne qu'occasionnent les bruits additionnels à un signal de parole et d'extraire en particulier l'information utile d'un signal d'observation bruité. De nombreuses approches de débruitage ont été proposées dans la littérature pour remédier à un tel problème [54-55].

Récemment, deux structures de séparation aveugle de sources dénommées directe et récursive ont été largement utilisées pour extraire la parole du bruit avec un minimum de dégradation. Nous traiterons dans ce chapitre le problème de la réduction du bruit acoustique et le rehaussement du signal de parole en proposant deux algorithmes bicapteurs basés sur la combinaison entre ces deux structures de séparation aveugle de sources, i.e. directe et récursive, et l'algorithme du filtrage adaptatif NLMS à convergence rapide (FNLMS).

Dans cette contribution, nous nous intéresserons à l'algorithme de filtrage adaptatif de type FNLMS (fast normalized least mean square). La particularité de cet algorithme est qu'il offre le meilleur des deux mondes: la propriété de convergence de l'algorithme FTF et la simplicité de calcul de l'algorithme NLMS.

L'organisation de ce chapitre est comme suit: dans la section 3.2 nous définirons le problème qui nous intéresse dans notre étude, ensuite la formulation mathématique des deux algorithmes proposés sera présentée dans la section 3.3, les initialisations de chaque algorithme seront données dans la section 3.4. La

complexité de calcul des deux algorithmes proposés sera évaluée dans la section 3.5. La section 3.6 sera réservée aux résultats expérimentaux des deux algorithmes proposés et finalement la conclusion sera donnée dans la section 3.7.

3.2. Position du problème

Le problème qui nous intéresse dans cette étude est le rehaussement du signal de parole pour les communications de type mains-libres dans une voiture.

Afin de modéliser au mieux l'environnement acoustique d'une communication de type mains-libres, à l'intérieur d'un habitacle de voiture, nous avons adopté le modèle du mélange linéaire convolutif simplifié, qui a été décrit dans la section 2.3.5.

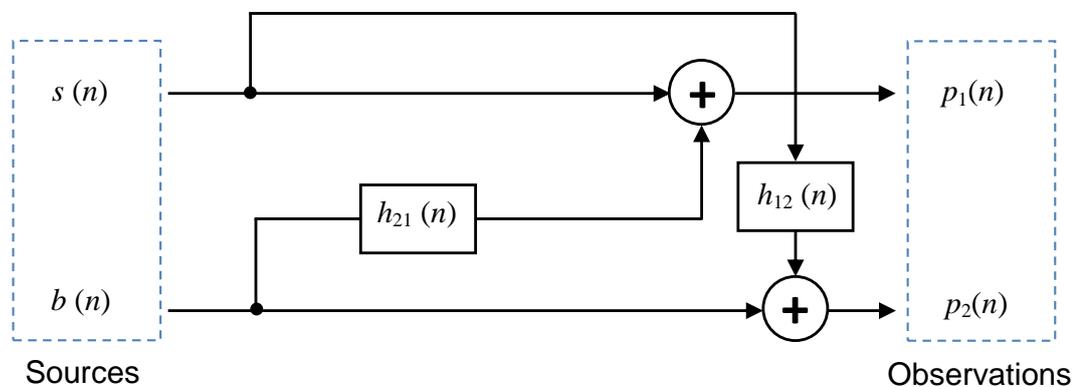


Figure 3. 1 : Schéma du modèle de mélange convolutif simplifié adopté.

La figure 3.1 représente le schéma du modèle de mélange convolutif simplifié, où, les signaux observés $p_1(n)$ et $p_2(n)$ sont générés par la propagation de deux signaux sources de parole et de bruit, ($s(n)$ et $b(n)$), respectivement, et qui sont supposés non corrélés. Le modèle de mélange décrit ci-dessus est basé sur deux filtres $h_{12}(n)$ et $h_{21}(n)$, qui représentent le couplage acoustique croisé entre les signaux sources et les microphones. Par conséquent les signaux d'observations $p_1(n)$ et $p_2(n)$ sont des mélanges convolutifs des deux signaux sources. Ces observations peuvent être modélisées comme suit:

$$p_1(n) = s(n) + h_{21}(n) * b(n) \quad (3.1)$$

$$p_2(n) = b(n) + h_{12}(n) * s(n) \quad (3.2)$$

Le problème principal est de récupérer les signaux sources $s(n)$ et $b(n)$ à partir des deux signaux observés $p_1(n)$ et $p_2(n)$, sans posséder aucune information sur les signaux sources. Pour résoudre ce problème, nous nous

sommes intéressés aux méthodes de réduction du bruit bicapteurs et en particulier les deux structures de séparation aveugle de sources de type directe et récursive.

3.3. Présentation des algorithmes bicapteurs proposés

Les structures de séparation aveugle de sources de types directe et récursive, sont fréquemment combinées avec différents algorithmes adaptatifs, pour améliorer la qualité et l'intelligibilité du signal de parole dans les systèmes de communication de type mains-libres [56-57].

Dans cette section, nous allons présenter deux algorithmes bicapteurs, basés sur la combinaison entre ces deux structures de séparation aveugle de sources et l'algorithme du filtrage adaptatif NLMS à convergence rapide (FNLMS). Ces deux algorithmes proposés héritent les bonnes caractéristiques de la séparation aveugle de sources et les bonnes propriétés de l'algorithme FNLMS, ce qui permet d'améliorer la qualité de débruitage du signal de parole.

3.3.1. Algorithme 1

Le premier algorithme proposé est basé sur la combinaison entre la structure de séparation aveugle de source directe (voir section 2.3.5) et l'algorithme FNLMS [58-59]. L'algorithme FNLMS est récemment proposé pour les applications d'annulation d'écho acoustique comme nous avons déjà vu dans le premier chapitre. Le choix de cet algorithme est motivé par sa complexité de calcul réduite ainsi que l'indépendance de ses performances de la nature du signal d'entrée. La figure 3.2 présente un schéma explicatif de l'Algorithme 1.

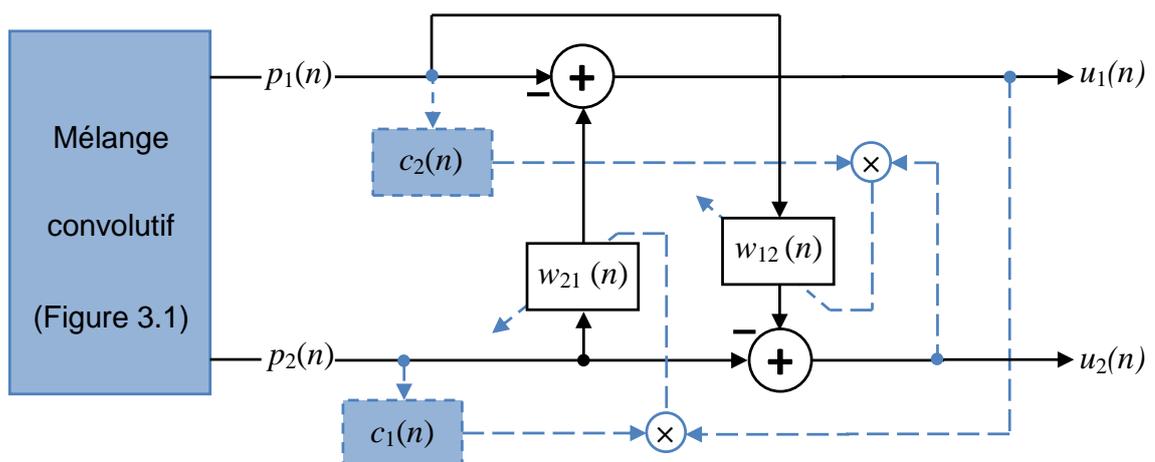


Figure 3. 2 : Schéma détaillé de l'Algorithme 1.

Les signaux de sortie $u_1(n)$ et $u_2(n)$ de l'Algorithme 1 sont donnés par les formules suivantes:

$$u_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{p}_2(n) \quad (3.3)$$

$$u_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{p}_1(n) \quad (3.4)$$

Où $\mathbf{p}_1(n) = [p_1(n), p_1(n-1), \dots, p_1(n-L+1)]^T$ et $\mathbf{p}_2(n) = [p_2(n), p_2(n-1), \dots, p_2(n-L+1)]^T$ sont les vecteurs qui contiennent les L derniers échantillons des signaux d'observations $p_1(n)$ et $p_2(n)$ respectivement. $\mathbf{w}_{21}(n) = [w_{21}(n), w_{21}(n-1), \dots, w_{21}(n-L+1)]^T$ et $\mathbf{w}_{12}(n) = [w_{12}(n), w_{12}(n-1), \dots, w_{12}(n-L+1)]^T$ sont des vecteurs contenant les coefficients des deux filtres adaptatifs $w_{21}(n)$ et $w_{12}(n)$ respectivement.

Les équations de mise à jour des deux filtres adaptatif $w_{21}(n)$ et $w_{12}(n)$ par l'Algorithme 1 sont données comme suit:

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) - \mu_1[u_1(n)\mathbf{c}_1(n)] \quad (3.5)$$

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) - \mu_2[u_2(n)\mathbf{c}_2(n)] \quad (3.6)$$

Où, μ_1 et μ_2 ($0 < \mu_1, \mu_2 < 2$) sont respectivement les pas d'adaptation des deux filtres adaptatifs $\mathbf{w}_{21}(n)$ et $\mathbf{w}_{12}(n)$.

Les vecteurs $\mathbf{c}_1(n)$ et $\mathbf{c}_2(n)$ sont les gains d'adaptation de l'Algorithme 1, définis par:

$$\mathbf{c}_1(n) = \gamma_1(n)\check{\mathbf{k}}_1(n) \quad (3.7)$$

$$\mathbf{c}_2(n) = \gamma_2(n)\check{\mathbf{k}}_2(n) \quad (3.8)$$

Où, $\gamma_1(n)$, $\gamma_2(n)$ et $\check{\mathbf{k}}_1(n)$, $\check{\mathbf{k}}_2(n)$ désignent respectivement les variables de vraisemblance et les gains de Kalman duaux.

Les variables de vraisemblance $\gamma_1(n)$ et $\gamma_2(n)$ de l'Algorithme 1 peuvent être calculées en utilisant la définition directe comme suit:

$$\gamma_1(n) = \frac{1}{1 - \check{\mathbf{k}}_1^T(n)\mathbf{p}_2(n)} \quad (3.9)$$

$$\gamma_2(n) = \frac{1}{1 - \check{\mathbf{k}}_2^T(n)\mathbf{p}_1(n)} \quad (3.10)$$

Les vecteurs gains de Kalman duaux $\check{\mathbf{k}}_1(n)$ et $\check{\mathbf{k}}_2(n)$ de l'Algorithme 1 sont donnés par:

$$\begin{bmatrix} \check{\mathbf{k}}_1(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_1(n)}{\lambda_f \alpha_1(n-1) + c_0} \\ \check{\mathbf{k}}_1(n-1) \end{bmatrix} \quad (3.11)$$

$$\begin{bmatrix} \check{\mathbf{k}}_2(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_2(n)}{\lambda_f \alpha_2(n-1) + c_0} \\ \check{\mathbf{k}}_2(n-1) \end{bmatrix} \quad (3.12)$$

Où, λ_f ($0 < \lambda_f < 1$) est un facteur d'oubli exponentiel et c_0 est une petite constante positive utilisée pour éviter la division par de très faibles valeurs en absence du signal d'entrée (périodes de silence). L'astérisque (*) dans (3.11) et (3.12) représente respectivement le dernier élément inutile des vecteurs $\check{\mathbf{k}}_1(n)$ et $\check{\mathbf{k}}_2(n)$.

$\alpha_1(n)$ et $\alpha_2(n)$ sont les variances des erreurs de prédiction aller, elles sont estimées récursivement par:

$$\alpha_1(n) = \lambda_f \alpha_1(n-1) + e_1^2(n) \quad (3.13)$$

$$\alpha_2(n) = \lambda_f \alpha_2(n-1) + e_2^2(n) \quad (3.14)$$

Les erreurs de prédiction aller $e_1(n)$ et $e_2(n)$ sont calculées à l'aide d'un prédicteur de premier ordre comme suit:

$$e_1(n) = p_2(n) - a_1 p_2(n-1) \quad (3.15)$$

$$e_2(n) = p_1(n) - a_2 p_1(n-1) \quad (3.16)$$

Où, a_1 et a_2 sont des paramètres de prédiction, ils sont obtenus en minimisant respectivement l'EQM $E[e_1^2(n)]$ et $E[e_2^2(n)]$. En annulant la dérivée de ces dernières fonctions par rapport à a_1 et a_2 respectivement, nous obtiendrons:

$$a_1 = \frac{E[p_2(n)p_2(n-1)]}{E[p_2^2(n-1)]} = \frac{r_1}{r_2} \quad (3.17)$$

$$a_2 = \frac{E[p_1(n)p_1(n-1)]}{E[p_1^2(n-1)]} = \frac{r_3}{r_4} \quad (3.18)$$

Où, r_1 et r_3 désignent respectivement, le premier coefficient de la fonction d'auto-corrélation des observations $p_2(n)$ et $p_1(n)$. r_2 et r_4 désignent respectivement, la puissance des signaux d'observations $p_2(n)$ et $p_1(n)$.

Les relations (3.17) et (3.18) sont évaluées par des estimateurs récursifs comme suit:

$$a_1(n) = \frac{r_1(n)}{r_2(n) + c_f} \quad (3.19)$$

$$a_2(n) = \frac{r_3(n)}{r_4(n) + c_f} \quad (3.20)$$

Avec:

$$r_1(n) = \eta_f r_1(n-1) + p_2(n) p_2(n-1) \quad (3.21)$$

$$r_2(n) = \eta_f r_2(n-1) + p_2^2(n) \quad (3.22)$$

$$r_3(n) = \eta_f r_3(n-1) + p_1(n) p_1(n-1) \quad (3.23)$$

$$r_4(n) = \eta_f r_4(n-1) + p_1^2(n) \quad (3.24)$$

Où, η_f ($0 < \eta_f < 1$) est un facteur d'oubli, et c_f est une constante de régularisation.

Les équations de fonctionnement de l'Algorithme 1 sont listées dans le tableau 3.1.

Tableau 3.1 : Listing du premier algorithme proposé (Algorithme 1).

Algorithme1
Etape de Prédiction:
Erreurs de prédiction: $e_1(n) = p_2(n) - a_1 p_2(n-1), e_2(n) = p_1(n) - a_2 p_1(n-1)$
Coefficients de prédiction: $a_1(n) = \frac{r_1(n)}{r_2(n)+c_f}, a_2(n) = \frac{r_3(n)}{r_4(n)+c_f}$ $r_1(n) = \eta_f r_1(n-1) + p_2(n) p_2(n-1), r_2(n) = \eta_f r_2(n-1) + p_2^2(n)$ $r_3(n) = \eta_f r_3(n-1) + p_1(n) p_1(n-1), r_4(n) = \eta_f r_4(n-1) + p_1^2(n)$
Variances des erreurs de prédiction: $\alpha_1(n) = \lambda_f \alpha_1(n-1) + e_1^2(n), \alpha_2(n) = \lambda_f \alpha_2(n-1) + e_2^2(n)$
Vecteurs gains de Kalman duaux: $\begin{bmatrix} \check{\mathbf{k}}_1(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_1(n)}{\lambda_f \alpha_1(n-1) + c_0} \\ \check{\mathbf{k}}_1(n-1) \end{bmatrix}$ $\begin{bmatrix} \check{\mathbf{k}}_2(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_2(n)}{\lambda_f \alpha_2(n-1) + c_0} \\ \check{\mathbf{k}}_2(n-1) \end{bmatrix}$
Variables de vraisemblance: $\gamma_1(n) = \frac{1}{1 - \check{\mathbf{k}}_1^T(n) \mathbf{p}_2(n)}, \gamma_2(n) = \frac{1}{1 - \check{\mathbf{k}}_2^T(n) \mathbf{p}_1(n)}$
Vecteurs gains d'adaptation: $\mathbf{c}_1(n) = \gamma_1(n) \check{\mathbf{k}}_1(n), \mathbf{c}_2(n) = \gamma_2(n) \check{\mathbf{k}}_2(n)$
Etape de filtrage:
Erreurs de filtrage <i>a priori</i> : $u_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1) \mathbf{p}_2(n), u_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1) \mathbf{p}_1(n)$
Etape d'adaptation:
Equations de mise à jour des filtres adaptatifs: $\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) - \mu_1 [u_1(n) \mathbf{c}_1(n)], \mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) - \mu_2 [u_2(n) \mathbf{c}_2(n)]$

3.3.2. Algorithme 2

Comme nous avons vu dans le chapitre 2 (section 2.3.5), l'inconvénient majeur de la structure de séparation aveugle de sources directe est la distorsion des signaux de sortie lorsque les microphones sont légèrement espacés. Pour surmonter cette contrainte nous avons proposé de combiner l'algorithme FNLMS avec la structure de séparation aveugle de sources récursive [60]. Cette combinaison suggérée (Algorithme 2) permettra une amélioration des performances de l'Algorithme 1 en termes de distorsion. La figure 3.3 présente un schéma détaillé de l'Algorithme 2.

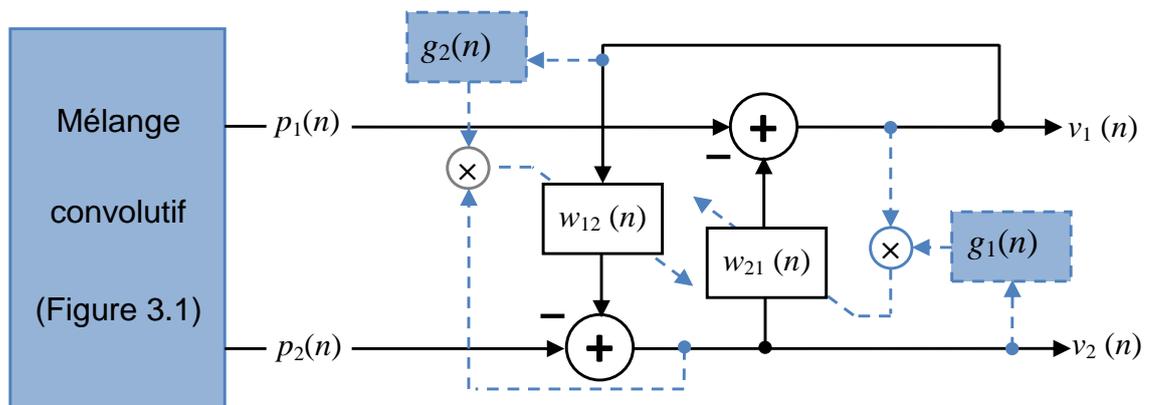


Figure 3. 3 : Schéma détaillé de l'Algorithme 2.

Les signaux de sortie $v_1(n)$ et $v_2(n)$ de l'Algorithme 2 sont donnés par les formules suivantes:

$$v_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1)\mathbf{v}_2(n) \quad (3.25)$$

$$v_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1)\mathbf{v}_1(n) \quad (3.26)$$

Où, $\mathbf{v}_1(n) = [v_1(n), v_1(n-1), \dots, v_1(n-L+1)]^T$ et $\mathbf{v}_2(n) = [v_2(n), v_2(n-1), \dots, v_2(n-L+1)]^T$ sont les vecteurs qui contiennent les L derniers échantillons des signaux de sortie $v_1(n)$ et $v_2(n)$ respectivement.

Les équations de mise à jour des deux filtres adaptatifs $w_{21}(n)$ et $w_{12}(n)$ par l'Algorithme 2 sont données comme suit:

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) - \mu_{21}[v_1(n)\mathbf{g}_1(n)] \quad (3.27)$$

$$\mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) - \mu_{12}[v_2(n)\mathbf{g}_2(n)] \quad (3.28)$$

Où, μ_{21} et μ_{12} ($0 < \mu_{21}, \mu_{12} < 2$) sont respectivement, les pas d'adaptation des deux filtres adaptatifs $\mathbf{w}_{21}(n)$ et $\mathbf{w}_{12}(n)$.

Les vecteurs $\mathbf{g}_1(n)$ et $\mathbf{g}_2(n)$ sont les gains d'adaptation de l'Algorithme 2, définis par:

$$\mathbf{g}_1(n) = \Gamma_1(n) \tilde{\mathbf{k}}_1(n) \quad (3.29)$$

$$\mathbf{g}_2(n) = \Gamma_2(n) \tilde{\mathbf{k}}_2(n) \quad (3.30)$$

Où, $\Gamma_1(n)$, $\Gamma_2(n)$ et $\tilde{\mathbf{k}}_1(n)$, $\tilde{\mathbf{k}}_2(n)$ désignent respectivement, les variables de vraisemblance et les gains de Kalman duaux.

Les variables de vraisemblance $\Gamma_1(n)$ et $\Gamma_2(n)$ de l'Algorithme 2 sont données par:

$$\Gamma_1(n) = \frac{1}{1 - \tilde{\mathbf{k}}_1^T(n) \mathbf{v}_2(n)} \quad (3.31)$$

$$\Gamma_2(n) = \frac{1}{1 - \tilde{\mathbf{k}}_2^T(n) \mathbf{v}_1(n)} \quad (3.32)$$

Les vecteurs gains de Kalman duaux $\tilde{\mathbf{k}}_1(n)$ et $\tilde{\mathbf{k}}_2(n)$ sont calculés comme dans l'Algorithme 1:

$$\begin{bmatrix} \tilde{\mathbf{k}}_1(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_1(n)}{\lambda_b a_1(n-1) + \xi} \\ \tilde{\mathbf{k}}_1(n-1) \end{bmatrix} \quad (3.33)$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_2(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_2(n)}{\lambda_b a_2(n-1) + \xi} \\ \tilde{\mathbf{k}}_2(n-1) \end{bmatrix} \quad (3.34)$$

Où, λ_b ($0 < \lambda_b < 1$) est un facteur d'oubli exponentiel et ξ est une petite constante positive utilisée pour éviter la division par de très faibles valeurs en absence du signal d'entrée (périodes de silence), l'astérisque (*) représente le dernier élément inutile des vecteurs $\tilde{\mathbf{k}}_1(n)$ et $\tilde{\mathbf{k}}_2(n)$.

Les variances des erreurs de prédiction aller $a_1(n)$ et $a_2(n)$ sont estimées récursivement comme suit:

$$a_1(n) = \lambda_b a_1(n-1) + \varepsilon_1^2(n) \quad (3.35)$$

$$a_2(n) = \lambda_b a_2(n-1) + \varepsilon_2^2(n) \quad (3.36)$$

Comme dans l'Algorithme 1, les erreurs de prédiction aller $\varepsilon_1(n)$ et $\varepsilon_2(n)$ sont calculées à l'aide d'un prédicteur de premier ordre comme suit:

$$\varepsilon_1(n) = v_2(n) - A_1 v_2(n-1) \quad (3.37)$$

$$\varepsilon_2(n) = v_1(n) - A_2 v_1(n-1) \quad (3.38)$$

Où, A_1 et A_2 sont des paramètres de prédiction. Ces paramètres se calculent de la même façon que dans l'Algorithme 1, i.e. en minimisant respectivement l'EQM $E[\varepsilon_1^2(n)]$ et $E[\varepsilon_2^2(n)]$, et on obtiendra:

$$A_1 = \frac{E[v_2(n)v_2(n-1)]}{E[v_2^2(n-1)]} = \frac{R_1}{R_2} \quad (3.39)$$

$$A_2 = \frac{E[v_1(n)v_1(n-1)]}{E[v_1^2(n-1)]} = \frac{R_3}{R_4} \quad (3.40)$$

Où, R_1 et R_3 désignent respectivement, le premier coefficient de la fonction d'auto-corrélation des signaux de sortie $v_2(n)$ et $v_1(n)$. R_2 et R_4 désignent respectivement, la puissance des signaux de sortie $v_2(n)$ et $v_1(n)$.

Les relations (3.39) et (3.40) sont évaluées par des estimateurs récursifs comme suit:

$$A_1(n) = \frac{R_1(n)}{R_2(n)+c_b} \quad (3.41)$$

$$A_2(n) = \frac{R_3(n)}{R_4(n)+c_b} \quad (3.42)$$

Avec:

$$R_1(n) = \eta_b R_1(n-1) + v_2(n) v_2(n-1) \quad (3.43)$$

$$R_2(n) = \eta_b R_2(n-1) + v_2^2(n) \quad (3.44)$$

$$R_3(n) = \eta_b R_3(n-1) + v_1(n) v_1(n-1) \quad (3.45)$$

$$R_4(n) = \eta_b R_4(n-1) + v_1^2(n) \quad (3.46)$$

Où, η_b ($0 < \eta_b < 1$) est un facteur d'oubli, et c_b est une constante de régularisation.

Les équations de fonctionnement de l'Algorithme 2 sont listées dans le tableau 3.2.

Tableau 3.2 : Listing du deuxième algorithme proposé (Algorithme 2).

Algorithme 2

Etape de Prédiction:

Erreurs de prédiction:

$$\varepsilon_1(n) = v_2(n) - A_1 v_2(n-1), \quad \varepsilon_2(n) = v_1(n) - A_2 v_1(n-1)$$

Coefficients de prédiction:

$$A_1(n) = \frac{R_1(n)}{R_2(n)+c_b}, \quad A_2(n) = \frac{R_3(n)}{R_4(n)+c_b}$$

$$R_1(n) = \eta_b R_1(n-1) + v_2(n) v_2(n-1), \quad R_2(n) = \eta_b R_2(n-1) + v_2^2(n)$$

$$R_3(n) = \eta_b R_3(n-1) + v_1(n) v_1(n-1), \quad R_4(n) = \eta_b R_4(n-1) + v_1^2(n)$$

Variances des erreurs de prédiction aller:

$$a_1(n) = \lambda_b a_1(n-1) + \varepsilon_1^2(n), \quad a_2(n) = \lambda_b a_2(n-1) + \varepsilon_2^2(n)$$

Vecteurs gains de Kalman duaux:

$$\begin{bmatrix} \tilde{\mathbf{k}}_1(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_1(n)}{\lambda_b a_1(n-1) + \xi} \\ \tilde{\mathbf{k}}_1(n-1) \end{bmatrix}, \quad \begin{bmatrix} \tilde{\mathbf{k}}_2(n) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_2(n)}{\lambda_b a_2(n-1) + \xi} \\ \tilde{\mathbf{k}}_2(n-1) \end{bmatrix}$$

Variables de vraisemblance:

$$\Gamma_1(n) = \frac{1}{1 - \tilde{\mathbf{k}}_1^T(n) v_2(n)}, \quad \Gamma_2(n) = \frac{1}{1 - \tilde{\mathbf{k}}_2^T(n) v_1(n)}$$

Vecteurs gains d'adaptation:

$$\mathbf{g}_1(n) = \Gamma_1(n) \tilde{\mathbf{k}}_1(n), \quad \mathbf{g}_2(n) = \Gamma_2(n) \tilde{\mathbf{k}}_2(n)$$

Etape de filtrage:

Erreurs de filtrage a priori:

$$v_1(n) = p_1(n) - \mathbf{w}_{21}^T(n-1) v_2(n), \quad v_2(n) = p_2(n) - \mathbf{w}_{12}^T(n-1) v_1(n)$$

Etape d'adaptation:

Equations de mise à jour des filtres adaptatifs:

$$\mathbf{w}_{21}(n) = \mathbf{w}_{21}(n-1) - \mu_{21} [v_1(n) \mathbf{g}_1(n)], \quad \mathbf{w}_{12}(n) = \mathbf{w}_{12}(n-1) - \mu_{12} [v_2(n) \mathbf{g}_2(n)]$$

3.4. Initialisation des algorithmes proposés

Les conditions initiales des deux algorithmes proposés (Algorithme 1 et Algorithme 2) sont les mêmes que celle de la version originale de l'algorithme FNLMS [28], i.e. nous avons appliqué les mêmes conditions initiales pour chaque algorithme proposé comme le montre le tableau 3.3.

Tableau 3.3 : Initialisation des algorithmes proposés: Algorithme1 et Algorithme 2.

Algorithmes proposés	Initialisation des paramètres
Algorithme 1	
Signaux d'entrée	$\mathbf{p}_1(0) = \mathbf{p}_2(0) = 0$
Filtres d'adaptation	$\mathbf{w}_{21}(0) = \mathbf{w}_{12}(0) = 0$
Gain d'adaptation	$\mathbf{c}_1(0) = \mathbf{c}_2(0) = 0$
Variances des erreurs de prédiction	$\alpha_1(0) = \alpha_2(0) = E_0$
Coefficients de prédiction	$r_1(0) = r_3(0) = 0$ et $r_2(0) = r_4(0) = E_0$
Algorithme 2	
Signaux de sortie	$\mathbf{v}_1(0) = \mathbf{v}_2(0) = 0$
Filtres d'adaptation	$\mathbf{w}_{21}(0) = \mathbf{w}_{12}(0) = 0$
Gain d'adaptation	$\mathbf{g}_1(0) = \mathbf{g}_2(0) = 0$
Variances des erreurs de prédiction	$a_1(0) = a_2(0) = E_0$
Coefficients de prédiction	$R_1(0) = R_3(0) = 0$ et $R_2(0) = R_4(0) = E_0$

Où, E_0 est une constante d'initialisation.

3.5. Evaluation de la complexité de calcul

Le tableau 3.4 résume la complexité de calcul (les multiplications et les divisions) des deux algorithmes proposés: Algorithme1 et Algorithme 2.

Tableau 3.4 : Complexité de calcul des deux algorithmes proposés
(Algorithme 1 et Algorithme 2).

Algorithmes	Nombre de Mult/Div
Algorithme 1	$6L + 26$
Algorithme 2	$6L + 26$

3.6. Résultats de simulation des algorithmes proposés

Dans cette section nous présenterons une validation expérimentale des deux algorithmes proposés (Algorithme 1 et Algorithme 2).

Nous décrirons, en premier lieu, les signaux tests utilisés en simulation à savoir:

- les différents signaux sources (parole et bruit),
- les réponses impulsionnelles.
- les signaux d'observations.

Nous présenterons dans un second temps, une étude comparative de chaque algorithme proposé avec d'autres algorithmes compétitifs. Cette étude sera réalisée en termes de plusieurs critères objectifs de qualité.

3.6.1. Description des signaux tests

▪ *Signaux sources*

Dans nos simulations nous avons utilisé deux signaux sources, la première source $s(n)$ est un signal de parole échantillonné à 8 kHz et d'environ 4 secondes prononcé par un locuteur masculin [61]. En ce qui concerne la seconde source $b(n)$, celle-ci correspond à un bruit USASI (United State of America Standard Institute), qui est un bruit stationnaire avec un spectre similaire au spectre moyen de la parole. Il est souvent utilisé comme signal de test dans le domaine de réduction du bruit acoustique pour évaluer la vitesse de convergence des algorithmes adaptatifs. Ces signaux sources $s(n)$ et $b(n)$ avec leurs spectrogrammes sont présentés respectivement par les figures 3.4 et 3.5.

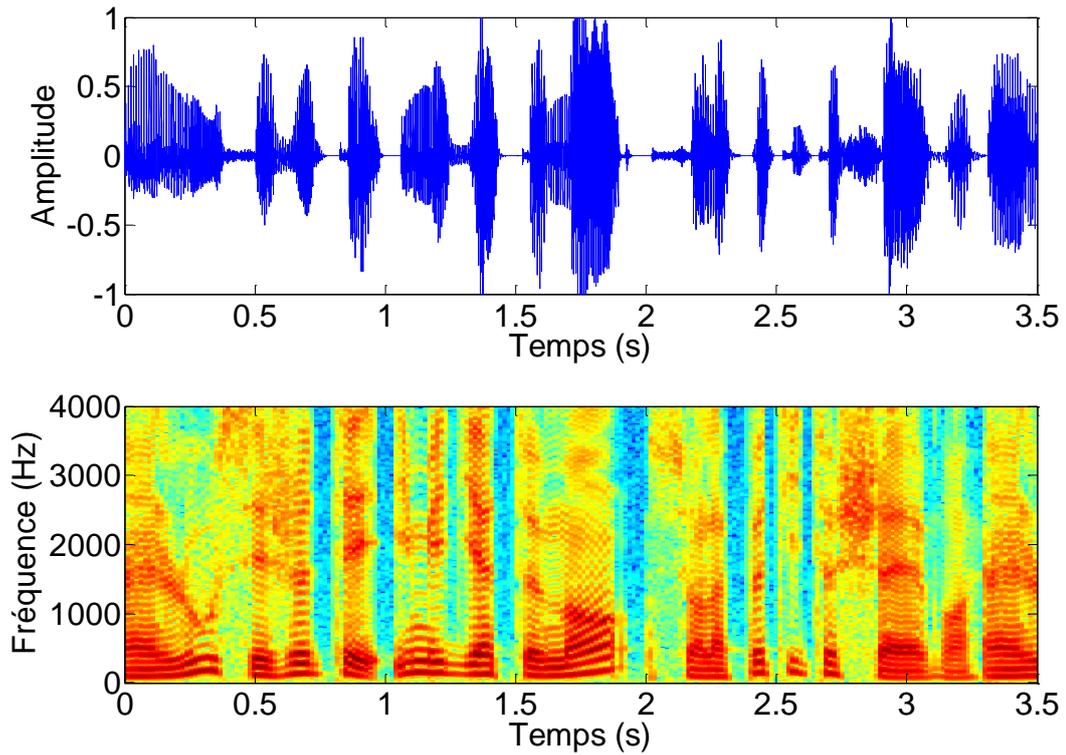


Figure 3. 4 : Signal de parole $s(n)$ avec son spectrogramme, $f_e = 8kHz$.

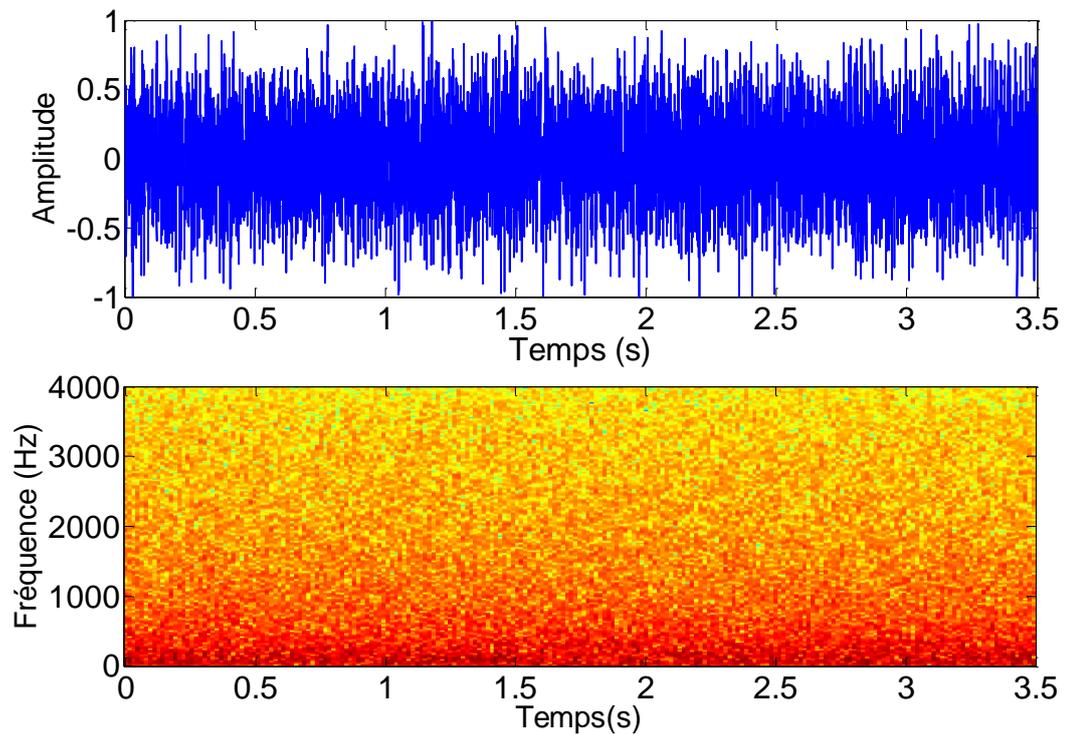


Figure 3. 5 : Signal de bruit USASI avec son spectrogramme, $f_e = 8kHz$.

- *Réponses impulsionnelles*

Afin de modéliser le couplage acoustique croisé entre les signaux sources et les microphones du modèle de mélange convolutif de la figure 3.1, nous

disposerons de deux réponses impulsionnelles $h_{12}(n)$ et $h_{21}(n)$. Ces deux réponses impulsionnelles sont construites selon [62] comme suit:

$$h_{12}(n) = \delta(n) + h'_{12}(n) \quad (3.47)$$

$$h_{21}(n) = \delta(n) + h'_{21}(n) \quad (3.48)$$

Où, $\delta(n)$ est une impulsion unitaire localisée au début de la réponse impulsionnelle, qui représente le trajet acoustique direct de chaque source et le microphone de couplage croisé.

h' est une queue de réponse impulsionnelle, qui représente la contribution des trajets multiples correspondant aux réflexions à l'intérieur de l'habitacle de voiture. Cette queue de réponse impulsionnelle est obtenue à partir d'un bruit aléatoire dont l'enveloppe temporelle est contrôlée selon la fonction de pondération suivante:

$$h(n) = Ae^{-Bn} \quad (3.49)$$

Où, A est un facteur d'échelle (dans notre cas, le scalaire A est pris égal à 1), et B est un facteur d'atténuation qui modélise l'absorption des ondes sonores sur les parois de la voiture et qui est donc lié au temps de réverbération tr . Ce facteur est donné par l'expression suivante:

$$B = 3 \log(10)/tr \quad (3.50)$$

Les amplitudes des réponses impulsionnelles sont ajustées à partir de la variance d'un bruit aléatoire qui est lié physiquement à l'espacement entre les deux microphones

Les figures (3.6) et (3.7) représentent respectivement un exemple des réponses impulsionnelles simulées $h_{12}(n)$ et $h_{21}(n)$ avec une taille égale à $L = 128$. La variance du bruit aléatoire est prise égale à 0.5 ce qui correspond à une configuration de microphones relativement espacés. Le facteur d'atténuation est pris égal à $B = 0.043$.

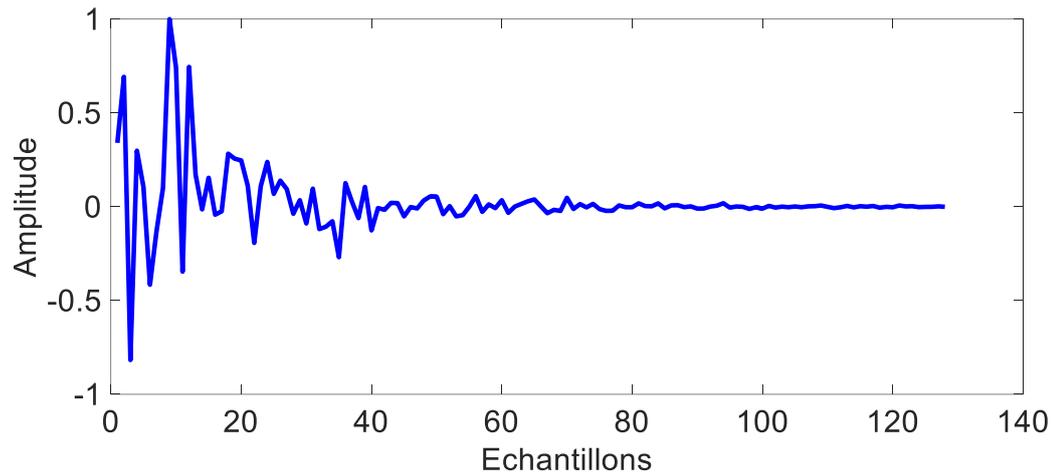


Figure 3. 6 : Exemple d'une réponse implusionnelle simulée $h_{21}(n)$, avec $L = 128$.

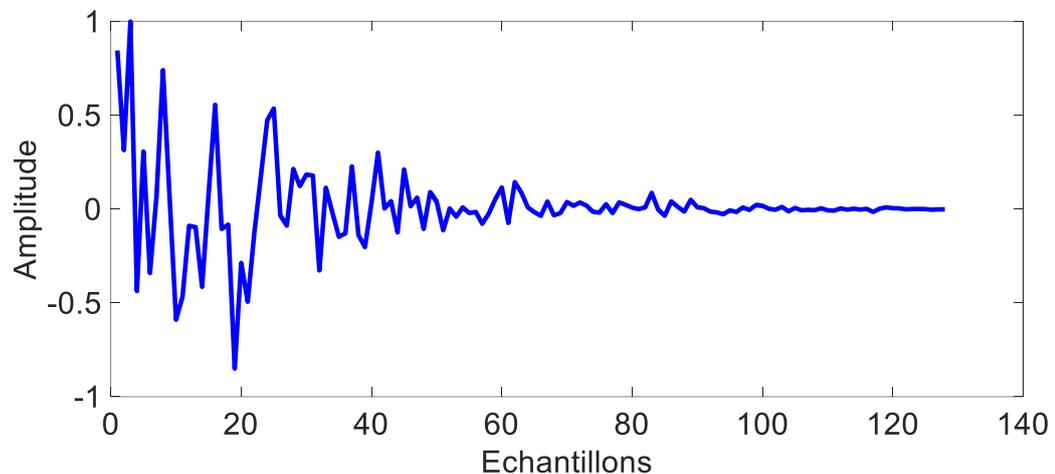


Figure 3. 7 : Exemple d'une réponse implusionnelle simulée $h_{12}(n)$, avec $L = 128$.

- *Signaux de mélange*

Nous avons utilisé les signaux décrits ci-dessus pour générer les deux signaux d'observation bruités $p_1(n)$ et $p_2(n)$, selon le modèle du mélange convolutif de la figure 3.1. Ces deux signaux de mélange sont les signaux d'entrée des deux algorithmes proposés.

Les figures 3.8 et 3.9 représentent respectivement les signaux observés $p_1(n)$ et $p_2(n)$ avec leurs spectrogrammes. Nous notons que le contexte de simulation des signaux d'observations est le suivant: le rapport signal à bruit (RSB) est fixé à 0dB sur les deux voix ($RSB_1 = RSB_2 = 0dB$).

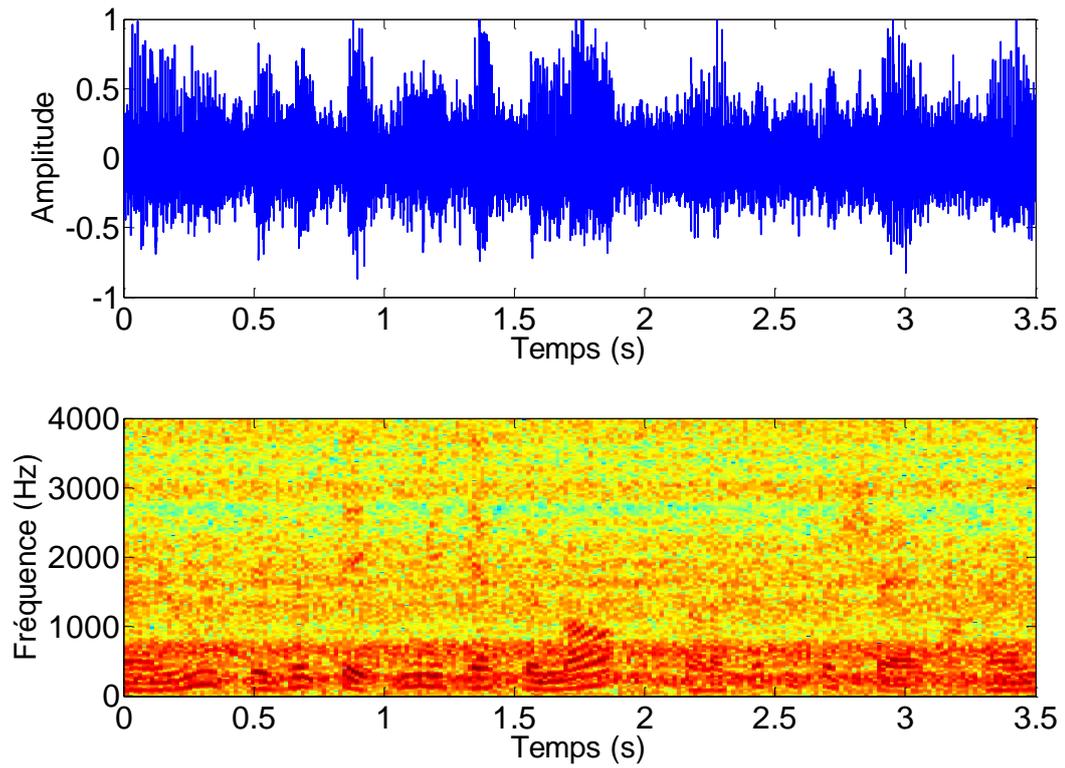


Figure 3. 8 : Signal observé $p_1(n)$ avec son spectrogramme.

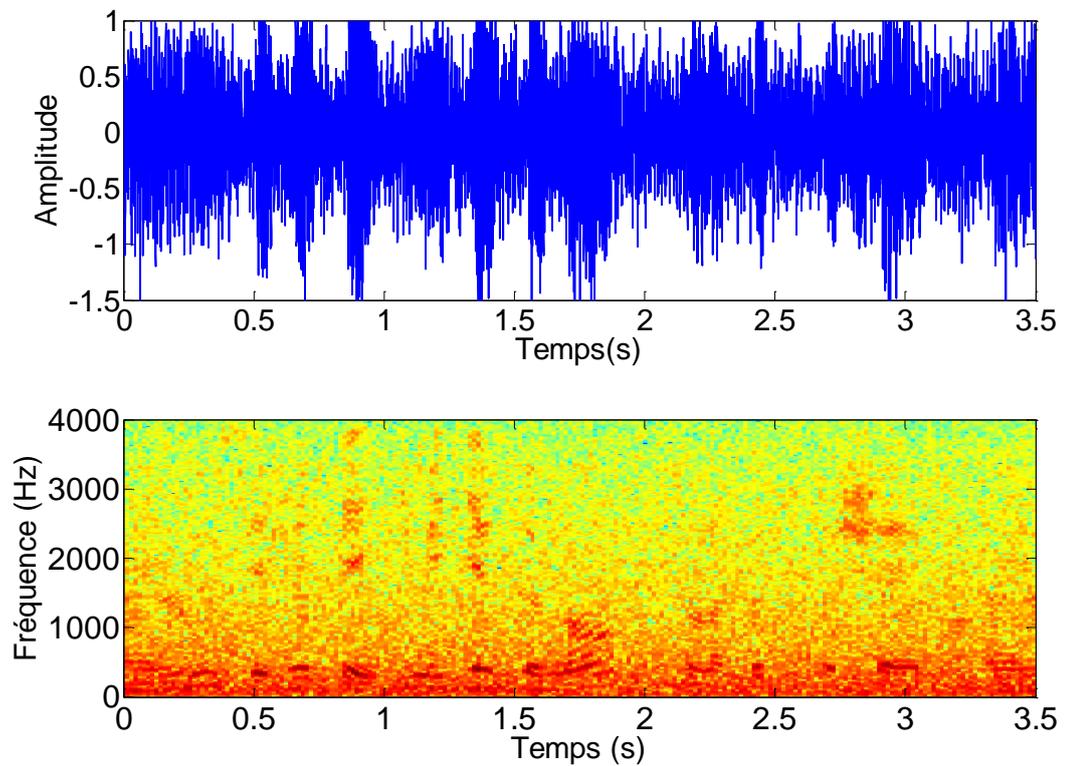


Figure 3. 9 : Signal observé $p_2(n)$ avec son spectrogramme.

3.6.2. Détection d'activité vocale

Dans notre étude, les deux algorithmes proposés (Algorithme 1 et Algorithme 2) utilisent un système de détection d'activité vocale (DAV) manuel pour contrôler l'adaptation des filtres $w_{12}(n)$ et $w_{21}(n)$. Cette adaptation est contrôlée de la manière suivante : le filtre $w_{21}(n)$ est adapté uniquement pendant les périodes d'inactivité vocale (période de présence de bruit seul), alors que le filtre $w_{12}(n)$ est adapté uniquement pendant les périodes d'activité vocale (période de présence du signal de parole). La figure 3.10 présente un exemple de DAV manuelle sur le signal de parole originale.

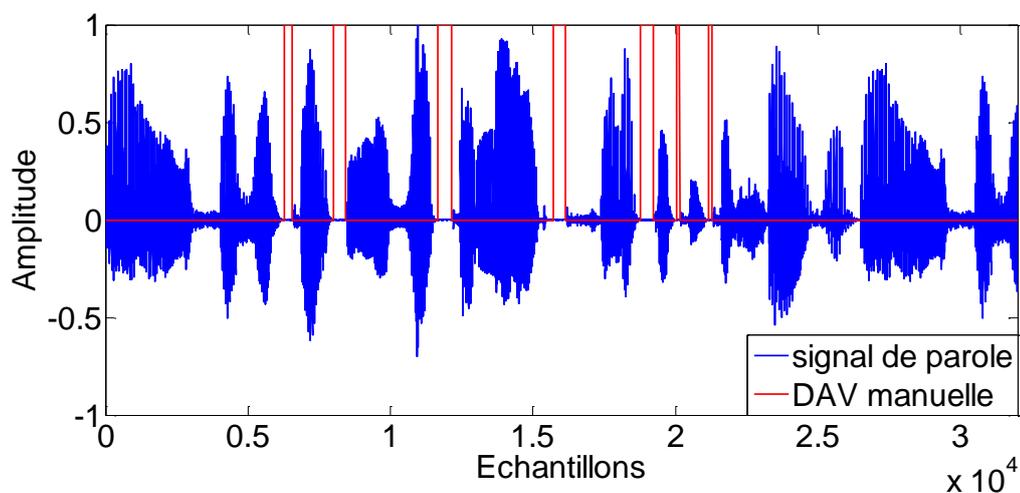


Figure 3. 10 : Signal de parole originale $s(n)$ avec sa segmentation (DAV) manuelle.

3.6.3. Description des critères de performance

Les mesures de performances des techniques de réduction du bruit se divisent en deux types: les mesures subjectives d'écoute avec un casque audio de qualité, et les mesures mathématiques objectives.

Les tests d'écoute subjectifs constituent la méthode la plus fiable pour évaluer la qualité ou l'intelligibilité du signal de parole. Cependant, ces tests sont coûteux en termes de temps et de ressources. Pour ces raisons, plusieurs chercheurs ont étudié la possibilité de concevoir des mesures objectives, basées sur une comparaison mathématique entre le signal de parole originale et rehaussé [63].

Afin d'évaluer les performances des deux algorithmes proposés, nous avons utilisé les mesures objectives de qualité suivantes:

- Critère du rapport signal à bruit segmental (RSB Seg).
- Critère du désajustement (SM).
- Critère de l'erreur quadratique moyenne segmental (EQM Seg).
- Critère de la distance cepstrale (DC).

- *Critère du Rapport Signal à Bruit Segmental (RSB Seg)*

Le RSB par segments temporels ou RSB segmental prend la moyenne du RSB de chaque trame utilisée dans le rehaussement, au lieu de prendre le RSB du signal au complet. Comme le rehaussement se fait sur une trame à la fois, cet indicateur permet de disposer d'une mesure de performance directe de la capacité d'un algorithme de réduire le bruit dans une trame [64]. Ce critère noté *RSBSeg* est défini par l'expression suivante:

$$RSBSeg_{dB} = \frac{10}{M} \sum_{m=0}^{M-1} \log_{10} \left(\frac{\sum_{n=Nm}^{Nm+N-1} |s(n)|^2}{\sum_{n=Nm}^{Nm+N-1} |s(n) - \hat{s}(n)|^2} \right) \quad (3.51)$$

Où $s(n)$ et $\hat{s}(n)$ sont respectivement les signaux de parole originale et rehaussé. Les paramètres M et N représentent respectivement le nombre de segments et la longueur de segment. Le symbole $|\cdot|$ représente l'opérateur absolu et le symbole \log_{10} est le logarithme en base 10 d'un nombre. Nous notons que tous les segments M correspondent aux périodes d'activité vocale.

- *Critère du Désajustement (SM)*

Le critère du désajustement est couramment utilisé pour évaluer les performances de vitesse de convergence des algorithmes de réduction du bruit. Ce dernier se calcule à partir de la distance euclidienne entre les coefficients des réponses impulsionnelles réelles et mesurées (estimées). Ce critère, noté SM pour 'System Mismatch', est défini par l'expression suivante [65]:

$$SM_{dB} = 10 \log_{10} \left(\frac{\|\mathbf{h} - \mathbf{w}\|^2}{\|\mathbf{h}\|^2} \right) \quad (3.52)$$

Où, \mathbf{h} représente le vecteur des coefficients du filtre réel et \mathbf{w} représente le vecteur des coefficients estimés. Le symbole $\|\cdot\|$ est l'opérateur de norme euclidienne.

- *Critère de l'Erreur Quadratique Moyenne Segmentale (EQM Seg)*

L'erreur quadratique moyenne segmentale est une mesure objective robuste dans l'évaluation des performances des algorithmes de réduction du bruit. Ce critère permet de quantifier le niveau du bruit résiduel en calculant la somme des

carrées de la différence entre le signal de parole originale et rehaussé, pendant les périodes d'inactivité vocale [66]:

$$EQMSe_{dB} = \frac{10}{M} \sum_{m=0}^{M-1} \log_{10} \left(\frac{1}{N} \sum_{n=Nm}^{N(m+1)-1} |s(n) - \hat{s}(n)|^2 \right) \quad (3.53)$$

Où, les paramètres M et N représentent respectivement le nombre de segments et la longueur de segment.

- *Critère de la Distance Cepstrale (DC)*

Le critère de la distance cepstrale est généralement utilisé pour quantifier la distorsion introduite par les systèmes de traitement de la parole. Ce critère est estimé en calculant la distance spectrale logarithmique entre le signal de parole originale et rehaussé. La DC est donnée par la relation suivante [67]:

$$CD_{dB} = \frac{10}{M} \sum_{m=0}^{M-1} \log_{10} \sum_{n=Tm}^{T(m+1)-1} (c_s(n) - c_{\hat{s}}(n))^2 \quad (3.54)$$

Où, $c_s(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log |S(\omega)| e^{j\omega n} d\omega$ et $c_{\hat{s}}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log |\hat{S}(\omega)| e^{j\omega n} d\omega$ sont respectivement, les nièmes coefficients cepstraux réels du signal de parole originale $s(n)$ et estimé $\hat{s}(n)$. $S(\omega)$ et $\hat{S}(\omega)$ représentent respectivement, les transformées de Fourier à court terme du signal de parole originale $s(n)$ et estimé $\hat{s}(n)$. T est la valeur moyenne de la distance cepstrale et M représente le nombre de segments pendant les périodes d'activité vocale.

3.6.4. Résultats de simulation de l'Algorithme 1

Cette section est consacrée à la présentation des résultats de simulation du premier algorithme proposé (Algorithme 1), qui correspond à une combinaison entre la structure FBSS et l'algorithme FNLMS. Des expériences intensives ont été menées pour évaluer les performances de l'Algorithme 1, en comparaison avec celles de l'algorithme NLMS bicapteurs (algorithme NLMS combiné avec la structure FBSS, dénommé NLMS 1) [52]. Cette étude comparative est basée sur l'évaluation des signaux de sortie et les critères objectifs de qualité qui ont été décrits dans la section précédente. Les paramètres de simulation de chaque algorithme sont résumés dans le tableau 3.5. Ces paramètres sont choisis afin d'optimiser les meilleures performances pour chaque algorithme simulé.

Notons que nous avons utilisé deux différents types de bruits pour générer les signaux d'observations $p_1(n)$ et $p_2(n)$. Un bruit blanc est utilisé pour tester les performances de stabilité de chaque algorithme simulé, et un bruit USASI est utilisé pour tester leurs performances de vitesse de convergence.

La taille des filtres réels et adaptatifs est choisie égale à $L = 128$. Nous avons varié le RSB d'entrée, en prenant deux valeurs, i.e. $RSB_1 = RSB_2 = 0,3dB$.

Comme nous nous intéresserons dans notre étude à la réduction du bruit et le rehaussement du signal de parole, dans les simulations ci-après, nous nous concentrerons uniquement à la sortie restituant le signal de parole, i.e. la première sortie de traitement $u_1(n)$, et le filtre adaptatif $w_{21}(n)$.

Tableau 3.5 : Paramètres de simulation de l'Algorithme 1 et NLMS 1.

Paramètres de simulation
Signaux d'entrées
Fréquence d'échantillonnage: $f_e = 8kHz$
RSB d'entrée: $RSB_1 = RSB_2 = 0,3dB$
Longueur des filtres réels: $L = 128$
Algorithme 1
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 1$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_f = 0.99$
Facteur d'oubli: $\eta_f = 0.9985$
Constantes positives: $c_o = 0.001, c_f = 1$
Constante d'initialisation: $E_0 = 0.5$
NLMS 1
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 1$
Longueur des filtres adaptatifs: $L = 128$

- *Evaluation des signaux de sortie*

Afin d'évaluer les performances de l'Algorithme 1 en comparaison avec l'algorithme NLMS 1, nous présenterons dans cette partie l'évolution temporelle du signal de parole originale, et celle des signaux de sortie obtenus avec les deux algorithmes, i.e. Algorithme 1 et NLMS 1. Les paramètres de simulation de chaque algorithme sont résumés dans le tableau 3.5.

A partir de la figure 3.11, nous pouvons remarquer que les signaux disponibles en sortie de traitement à partir des deux algorithmes, Algorithme 1 et NLMS 1, sont visuellement nettoyés du bruit et très proches du signal de parole originale.

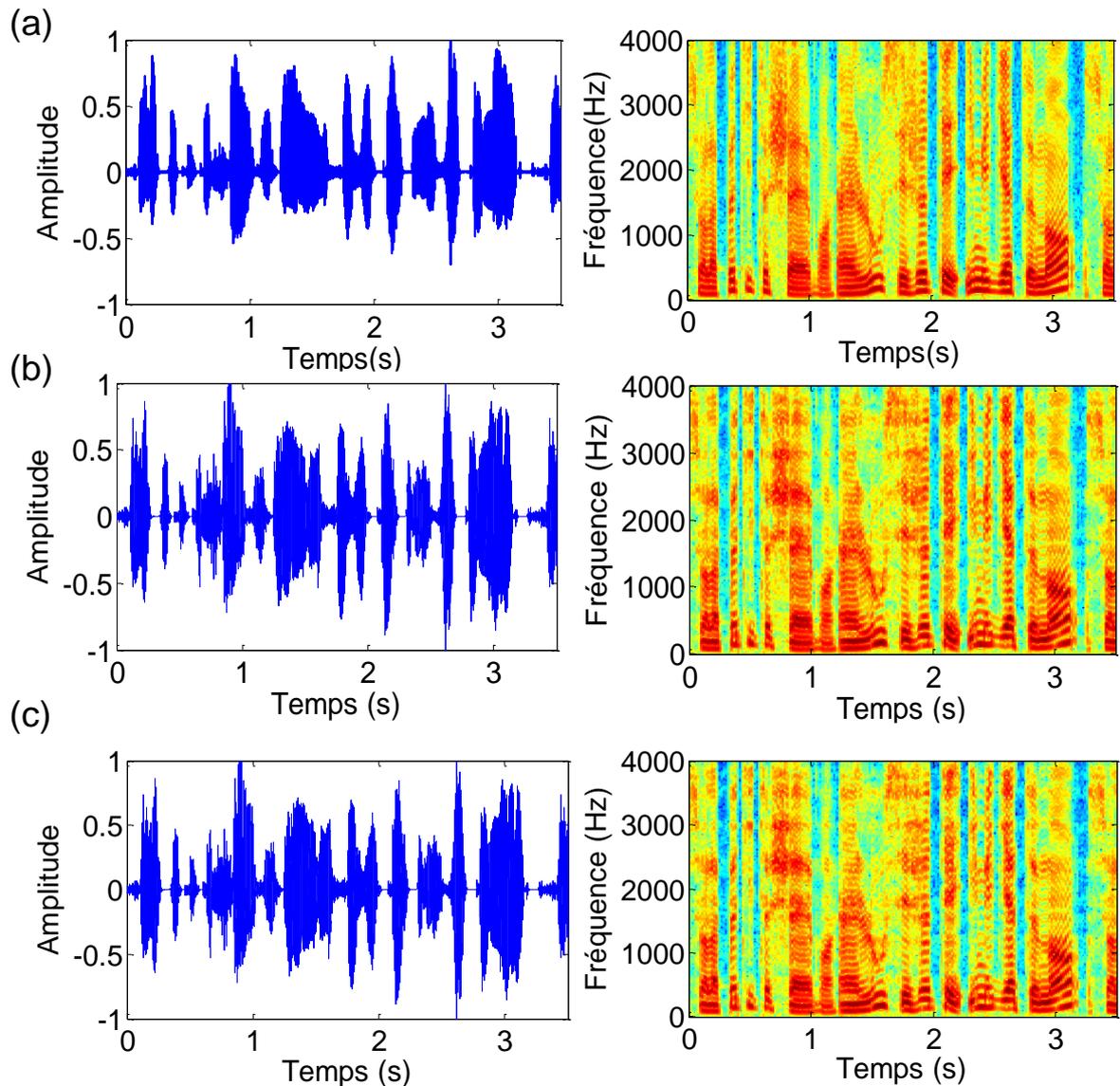


Figure 3. 11 :Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par NLMS 1, (c) signal de sortie obtenu par l'Algorithme 1.

- *Evaluation du rapport signal à bruit segmental*

Dans cette simulation, nous avons évalué les performances de réduction du bruit de l'algorithme 1 proposé en comparaison avec l'algorithme NLMS 1, en termes du critère rapport signal à bruit segmental. Les paramètres de simulation de chaque algorithme sont résumés dans le tableau 3.5.

Les résultats comparatifs de l'évolution du rapport signal à bruit segmental à la sortie des deux algorithmes (i.e. Algorithme 1 et NLMS 1) pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0, 3dB$, et deux différents types de bruits, i.e. USASI et blanc, sont reportés sur les figures 3.12 et 3.13.

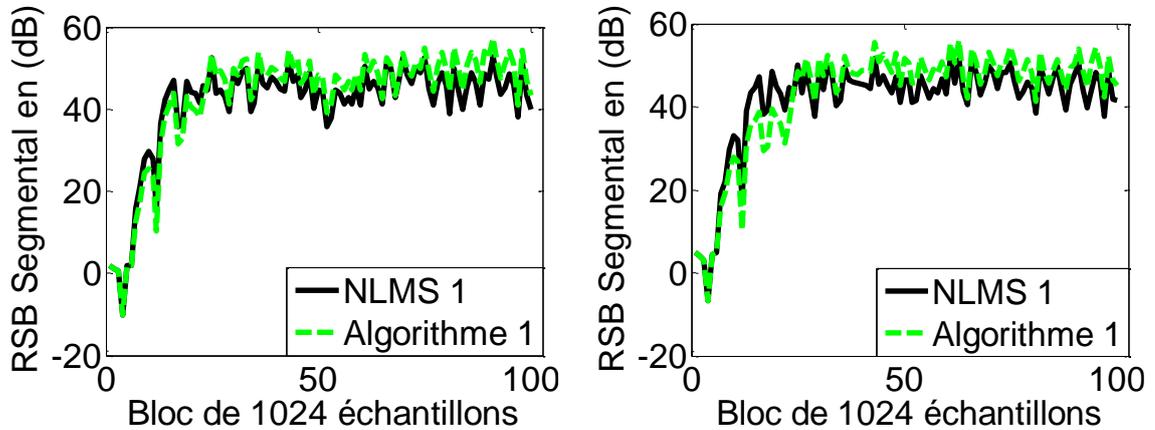


Figure 3. 12 : Evaluation du critère RSB segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

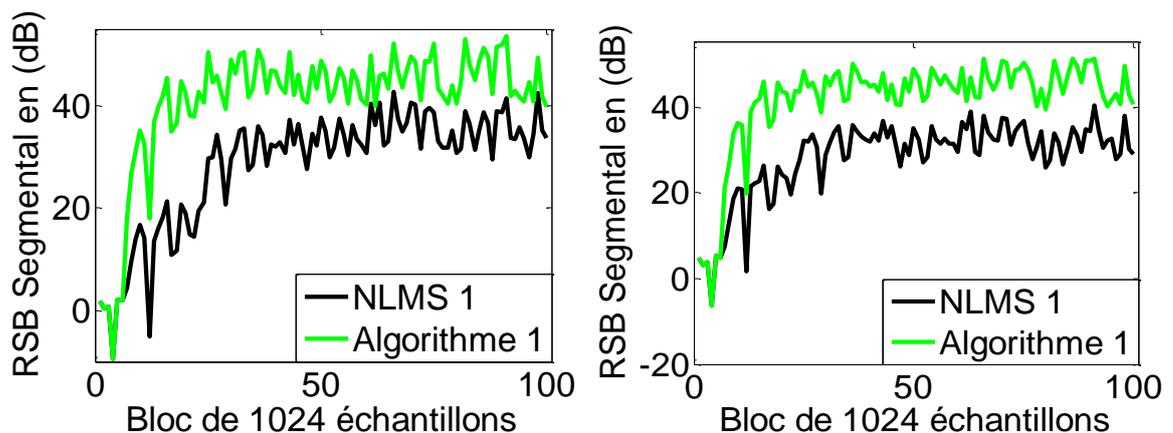


Figure 3. 13 : Evaluation du critère RSB segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 = RSB_2 = 3dB$.

A partir de ces résultats, un comportement similaire est noté pour l'algorithme 1 proposé et l'algorithme NLMS 1 lorsque le bruit blanc est utilisé. Cependant nous pouvons bien remarquer le bon comportement de l'algorithme 1 proposé par rapport à l'algorithme NLMS 1 avec les composantes du bruit USASI. Les mêmes remarques sont notées lorsque les signaux d'entrée sont fortement altérés par les composantes du bruit acoustique, i.e. $RSB_1 = RSB_2 = 0dB$. Ces

résultats prouvent que l'algorithme 1 proposé réduit le bruit acoustique d'une manière plus efficace que l'algorithme NLMS 1 dans différentes situations, où des bruits corrélés, et non corrélés sont présents.

- *Evaluation du désajustement*

Afin d'analyser les performances de vitesse convergence de l'algorithme 1 proposé en comparaison avec l'algorithme NLMS 1, nous présenterons sur les figures 3.14 et 3.15 l'évolution temporelle du désajustement mesuré sur les coefficients du filtre adaptatif $w_{21}(n)$, pour les deux algorithmes, i.e. Algorithme 1 et NLMS 1. Notons que Les paramètres de simulation sont les mêmes que ceux indiqués dans le tableau 3.5.

Dans l'ensemble de simulation présentée dans cette partie, nous avons évalué le critère du désajustement pour deux niveaux de RSB d'entrée i.e. $RSB_1 = RSB_2 = 0,3dB$, et deux types de bruits, i.e. USASI et blanc. D'après les résultats obtenus, nous pouvons remarquer que dans le cas du bruit USASI, l'algorithme 1 proposé converge plus rapidement vers la solution optimale par rapport à l'algorithme NLMS 1, qui est fortement pénalisé par les propriétés du signal d'entrée. Nous avons également noté que dans le cas du bruit blanc, l'algorithme 1 proposé et l'algorithme NLMS 1 fournissent presque les mêmes performances en termes de vitesse de convergence.

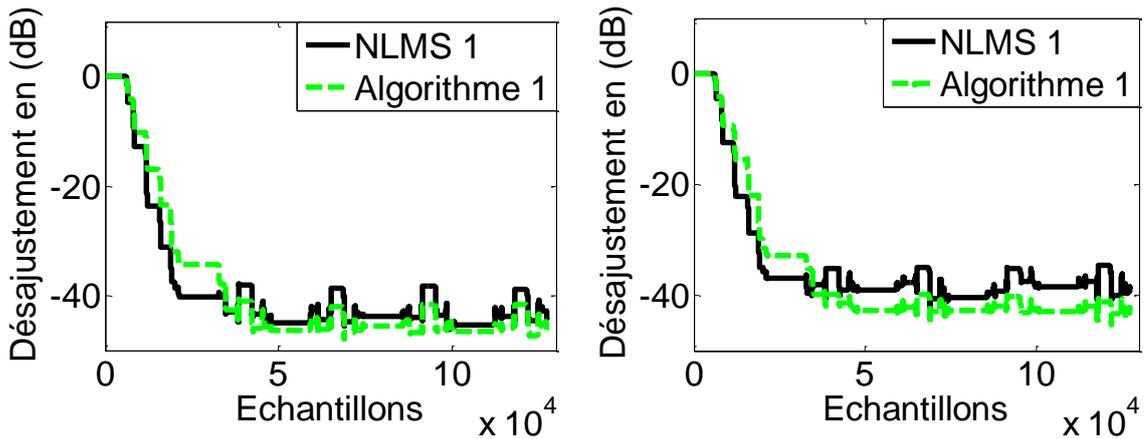


Figure 3. 14 : Evaluation du critère de désajustement pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

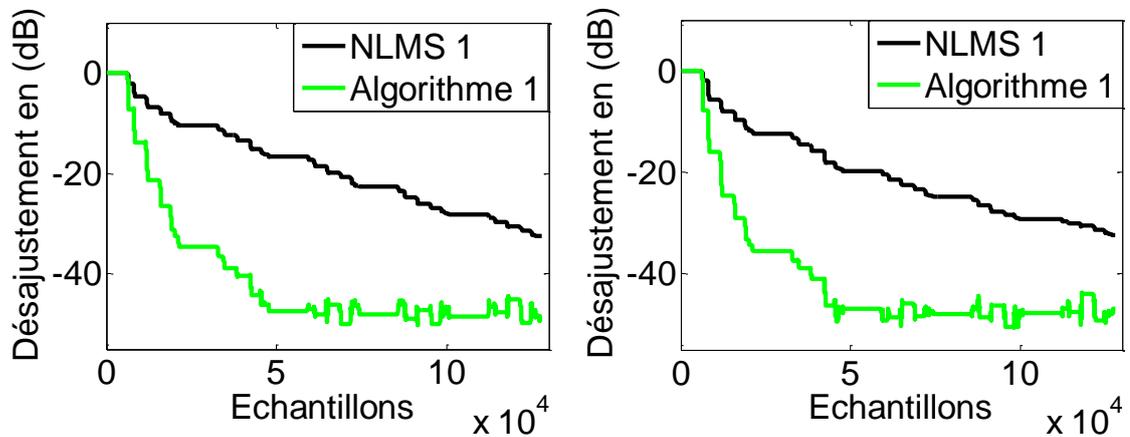


Figure 3. 15 : Evaluation du critère de désajustement pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de l'erreur quadratique moyenne segmentale*

De manière à compléter l'analyse de comportement de vitesse de convergence de l'algorithme 1 proposé en comparaison avec l'algorithme NLMS 1, nous avons évalué dans cette partie, le critère de l'erreur quadratique moyenne segmental (EQM Seg).

Les figures 3.16 et 3.17 présentent les résultats de l'évaluation du critère de l'EQM Seg obtenus avec les deux algorithmes, i.e. Algorithme 1 proposé et NLMS 1, ceci pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0, 3dB$, et deux types de bruits, i.e. USASI et blanc. Nous notons que les paramètres de

simulation de chaque algorithme sont les mêmes que ceux donnés par le tableau 3.5.

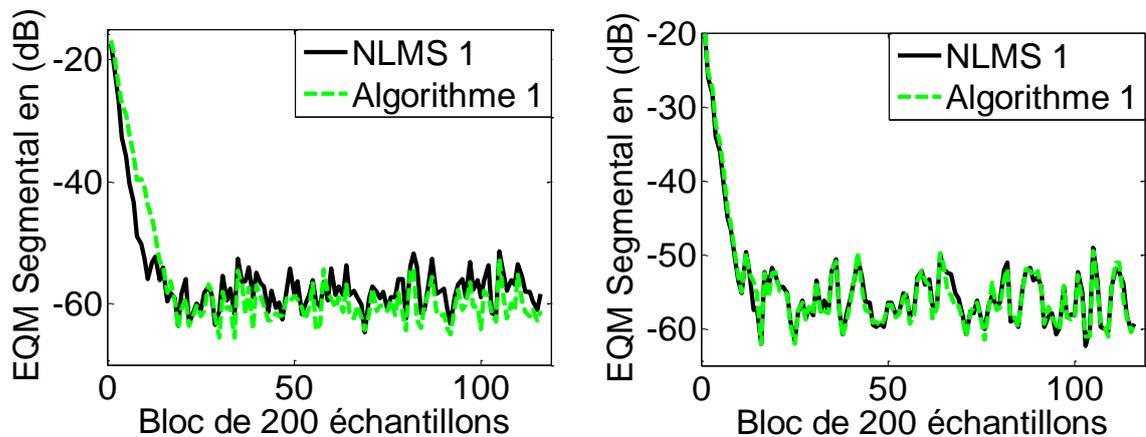


Figure 3. 16 : Evaluation du critère EQM segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB

d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$

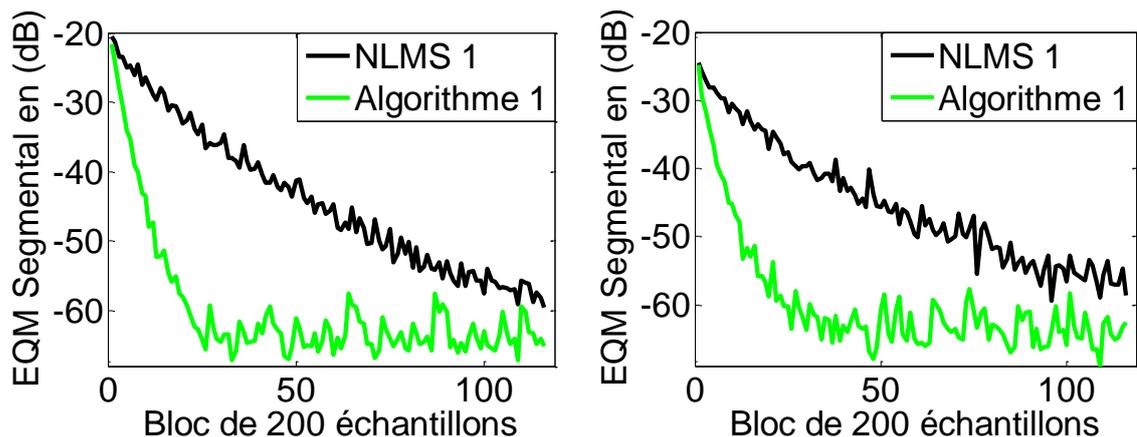


Figure 3. 17 :Evaluation du critère EQM segmental pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB

d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

D'après ces résultats, nous pouvons clairement observer dans le cas du bruit USASI, la supériorité en performance de vitesse de convergence de l'algorithme 1 proposé par rapport à l'algorithme NLMS 1. Cette bonne propriété de vitesse de convergence rapide est due au fait que l'algorithme 1 proposé utilise l'erreur de prédiction au lieu du signal d'entrée dans la génération du gain d'adaptation, ce qui permet de blanchir le signal d'entrée. Nous notons également un comportement similaire pour l'algorithme proposé et l'algorithme NLMS 1 lorsque le bruit blanc est utilisé.

Finalement, nous concluons que l'algorithme proposé offre une bonne propriété de vitesse de convergence dans différentes conditions, i.e. observations légèrement ou fortement bruités, et situations, i.e. composantes de bruits corrélées ou non corrélées.

- *Evaluation de la distance cepstrale*

Dans cette dernière simulation, nous avons évalué le critère de la distance cepstrale pour mesurer la distorsion apportée par l'algorithme 1 proposé et l'algorithme NLMS 1.

Nous avons estimé le critère de la DC pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0,3dB$, et deux différents types de bruits, i.e. USASI et blanc. Les paramètres de simulation de chaque algorithme sont similaires à ceux des expériences précédentes et sont également résumés dans le tableau 3.5. Les résultats de simulation sont reportés sur les figures 3.18 et 3.19.

Les résultats obtenus dans le cas du bruit blanc montrent que l'algorithme 1 proposé et l'algorithme NLMS 1 ont le même comportement en termes de la distance cepstrale. Cependant dans le cas du bruit USASI nous pouvons bien remarquer que le signal de parole obtenu en sortie de traitement de l'algorithme 1 proposé est moins distordu par rapport à celui de l'algorithme NLMS 1. Nous notons également que l'algorithme NLMS 1 montre un faible comportement lorsque les signaux d'observations sont fortement altérés par les composantes du bruit acoustique, i.e. $RSB_1 = RSB_2 = 0dB$. Ces résultats confirment que l'algorithme 1 proposé réduit le bruit acoustique en sortie avec un minimum de distorsion.

Finalement, à travers tous les résultats de simulation effectués avec l'algorithme 1, nous avons confirmé l'efficacité de ce dernier dans le domaine de débruitage de la parole.

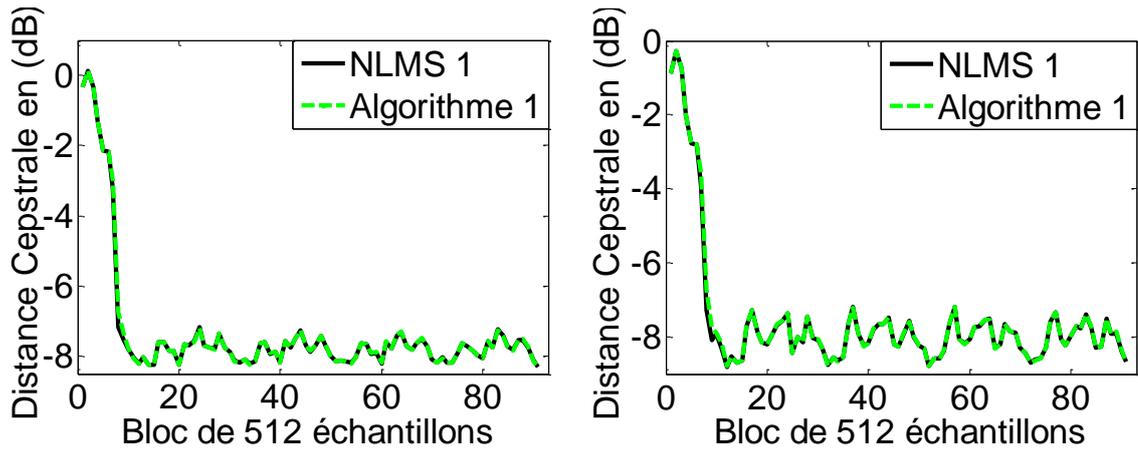


Figure 3. 18 : Evaluation du critère de la DC pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

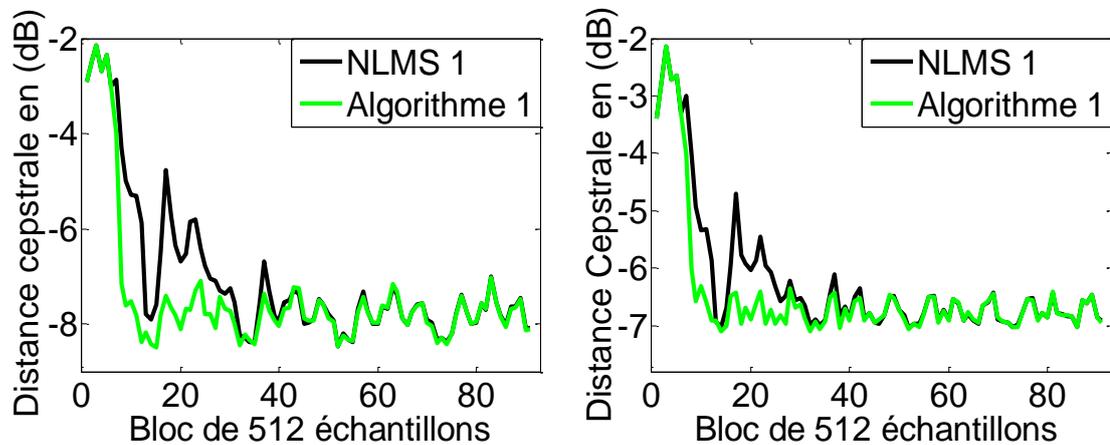


Figure 3. 19 : Evaluation du critère de la DC pour l'algorithme 1 proposé et l'algorithme NLMS 1, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

3.6.5. Résultat de simulation de l'Algorithme 2

Dans cette section, nous examinerons les performances du deuxième algorithme proposé (Algorithme 2) en comparaison avec celles de l'algorithme NLMS 2 bicapteurs, qui correspond à une combinaison entre la structure BBSS et l'algorithme NLMS [52]. Nous avons effectué plusieurs simulations en termes des critères objectifs de qualité suivants:

- (i). L'évaluation temporelle des signaux de sortie;
- (ii). Le rapport signal à bruit segmental;
- (iii). Le désajustement;
- (iv). L'erreur quadratique moyenne segmentale;

(v). La distance cepstrale.

Les paramètres de simulation de chaque algorithme sont résumés dans le tableau 3.6. Notons que les signaux d'entrée et les réponses impulsionnelles utilisés pour générer les signaux de mélange correspondent à ceux présentés dans la section 3.6.1.

Nous avons également repris la même démarche d'étude et de simulation que celle de l'Algorithme 1.

Tableau 3.6 : Paramètres de simulation de l'Algorithme 2 et NLMS 2.

Paramètres de simulation
Signaux d'entrées
Fréquence d'échantillonnage: $f_e = 8kHz$
RSB d'entrée: $RSB_1 = RSB_2 = 0,3dB$
Longueur des filtres réels: $L = 128$
Algorithme 2
Pas d'adaptation des filtres adaptatifs: $\mu_{12} = \mu_{21} = 1$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_b = 0.99$
Facteur d'oubli: $\eta_b = 0.9985$
Constantes positives: $\xi = 0.001, c_b = 1$
Constante d'initialisation: $E_0 = 0.5$
NLMS 2
Pas d'adaptation des filtres adaptatifs: $\mu_{12} = \mu_{21} = 1$
Longueur des filtres adaptatifs: $L = 128$

- *Evaluation des signaux de sortie*

Dans cette partie, un simple test visuel sur les signaux de sortie obtenus avec les deux algorithmes, i.e. Algorithme 2 et NLMS 2 est présenté. Les paramètres de simulation de chaque algorithme sont similaires à ceux résumés dans le tableau 3.6. Notons que nous avons utilisé le bruit USASI comme source

de bruit ponctuelle, et le RSB d'entrée est fixé à $0dB$ sur les deux voies, i.e. $RSB_1 = RSB_2 = 0dB$.

Nous présenterons sur la figure 3.20, l'évolution temporelle du signal de parole originale et celle des signaux de sortie obtenus avec les deux algorithmes, i.e. Algorithme 2 et NLMS 2. Pour chaque algorithme, nous avons également présenté le spectrogramme du signal disponible en sortie de traitement. A partir de cette figure, nous pouvons remarquer que les signaux disponibles en sortie de traitement à partir des deux algorithmes sont presque totalement débruités, cela est bien prouvé en examinant les spectrogrammes des signaux de sortie.

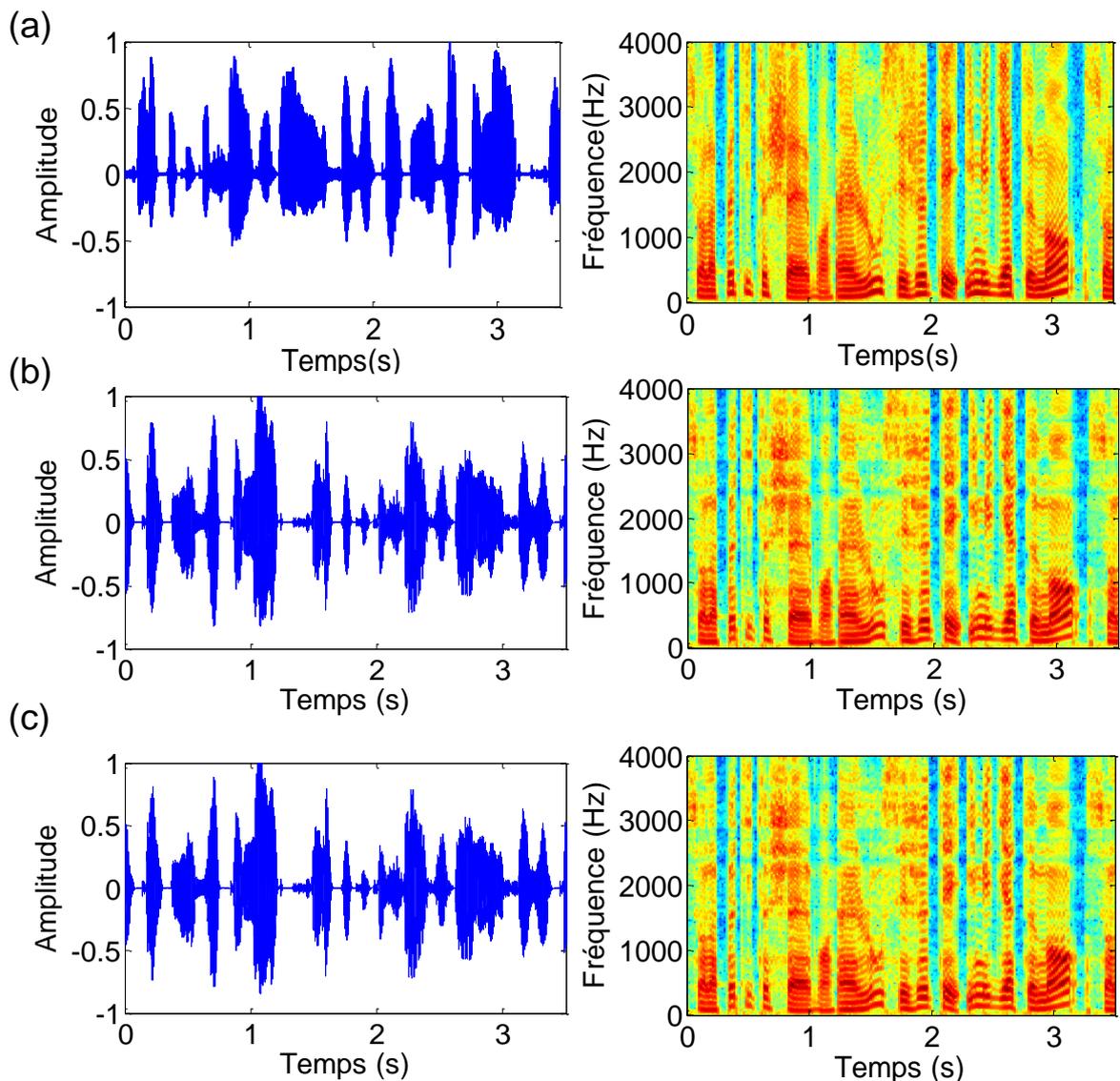


Figure 3. 20 : Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par NLMS 2, (c) signal de sortie obtenu par l'Algorithme 2.

- *Evaluation du rapport signal à bruit segmental*

Dans cette simulation, nous avons évalué le critère du rapport signal à bruit segmental pour les deux algorithmes, i.e. Algorithme 2 et NLMS 2. Les résultats obtenus pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$, et deux différents types de bruits, i.e. USASI et blanc, sont reportés sur les figures 3.21 et 3.22. Les paramètres de simulation de chaque algorithme sont similaires à ceux résumés dans le tableau 3.6.

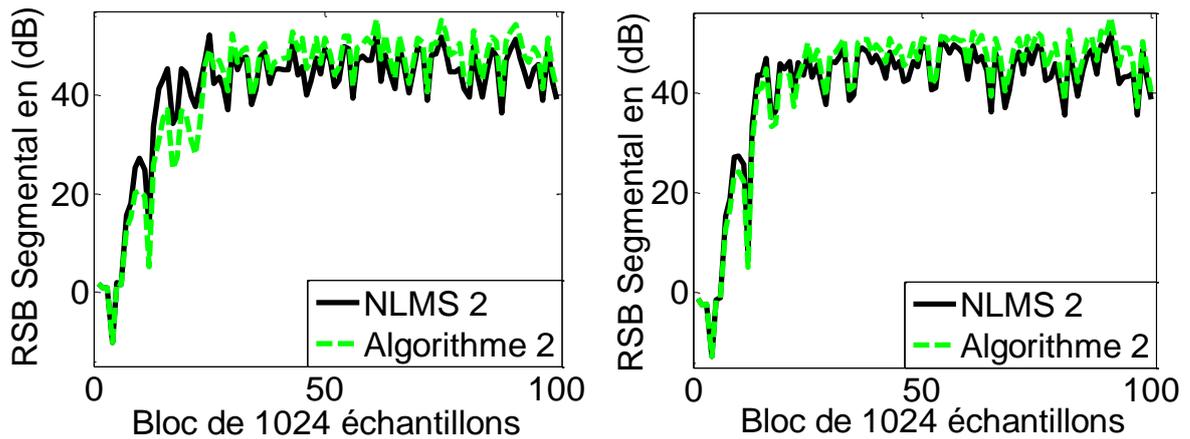


Figure 3. 21 : Evaluation du critère RSB segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

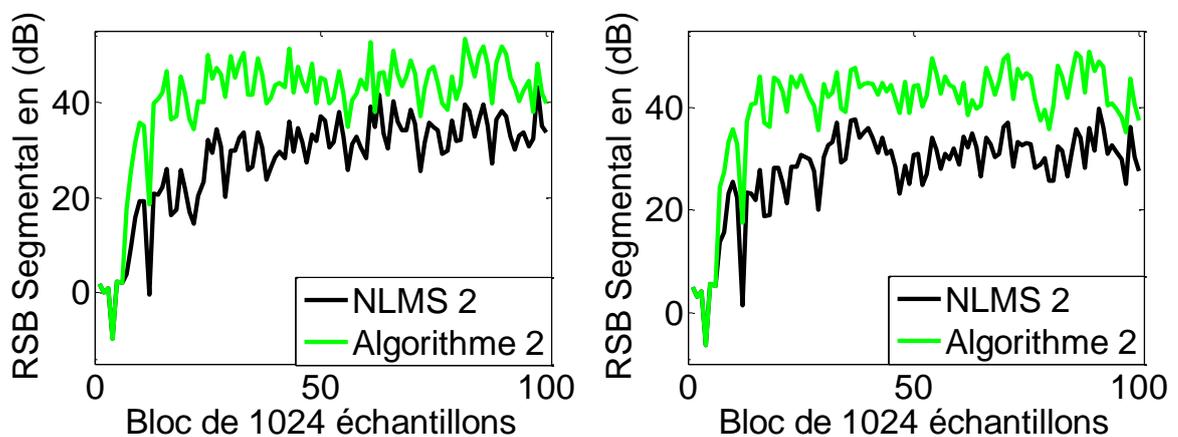


Figure 3. 22 : Evaluation du critère RSB segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

L'analyse des résultats de simulation obtenus dans le cas du bruit blanc montre bien que les deux algorithmes Algorithme 2 et NLMS 2 possèdent le même comportement en termes du RSB segmental. En revanche, dans le cas du bruit

USASI, les valeurs du RSB segmental obtenues à la sortie de l'algorithme 2 proposé sont plus importantes que celles d'algorithme NLMS 2. Cela s'explique par l'indépendance des performances de l'algorithme 2 proposé de la nature du signal d'entrée.

▪ *Evaluation du désajustement*

Nous présenterons sur les figures 3.23 et 3.24 l'évolution temporelle du désajustement mesuré sur les coefficients du filtre adaptatif $w_{21}(n)$ des deux algorithmes (Algorithme 2 et NLMS 2), ceci pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$, et deux différents types de bruits, i.e. USASI et blanc.

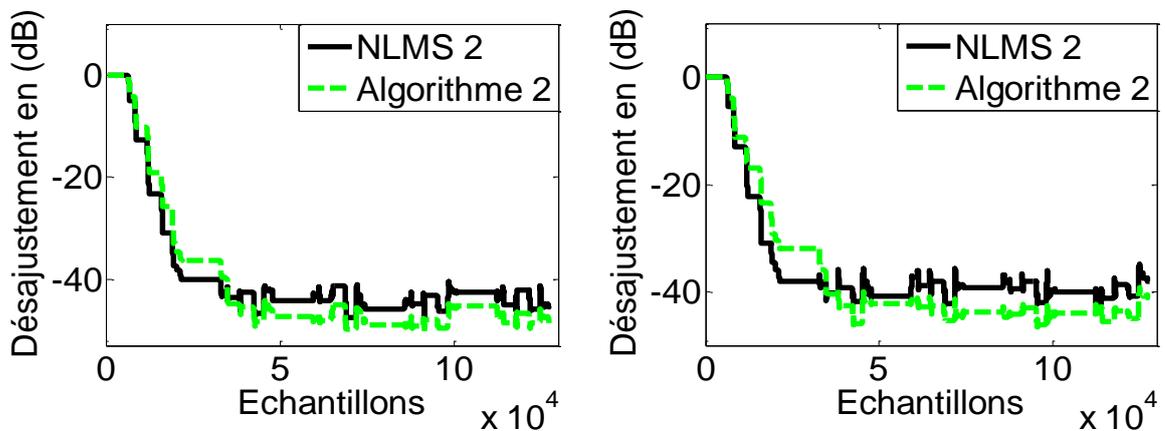


Figure 3. 23 : Evaluation du critère de désajustement pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

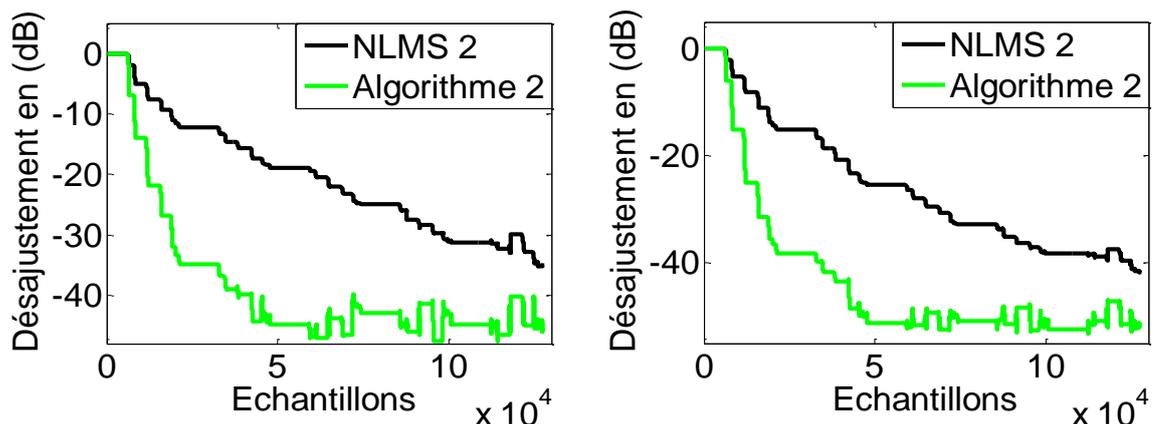


Figure 3. 24 : Evaluation du critère de désajustement pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

D'après les résultats obtenus dans cette expérience, nous notons une similitude des performances des deux algorithmes (Algorithme 2 et NLMS 2) avec

les composantes du bruit blanc. Nous remarquons également la bonne performance de vitesse de convergence de l'algorithme 2 en comparaison avec l'algorithme NLMS 2, lorsque le bruit USASI est présent. En outre, l'algorithme 2 garde une vitesse de convergence rapide même dans un scénario très bruyant, i.e. $RSB_1 = RSB_2 = 0 \text{ dB}$.

Dans l'ensemble de tests, l'algorithme NLMS 2 montre un faible comportement en termes de vitesse de convergence avec les composantes du bruit USASI, ceci est dû principalement à sa dépendance des propriétés du signal d'entrée.

- *Evaluation de l'erreur quadratique moyenne segmentale*

Pour compléter les résultats précédents, nous montrerons sur les figures 3.25 et 3.26 la comparaison des performances de vitesse de convergence des deux algorithmes, i.e. Algorithmes 2 et NLMS 2, en termes du critère de l'EQM segmental.

Dans cette simulation, nous avons repris la même démarche que celle des sections précédentes, où nous avons évalué l'EQM segmental pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et 3 dB , et deux type de bruits, i.e. USASI et blanc. Les résultats obtenus confirment à nouveaux qu'en ayant des bruits corrélés à l'entrée, l'algorithme 2 présente de meilleures performances de vitesse de convergence devant l'algorithme NLMS 2.

Les résultats de l'évaluation du désajustement et d'EQM segmentale prouvent que l'algorithme 2 converge vers la solution souhaitée d'une manière plus stable et rapide que l'algorithme NLMS 2.

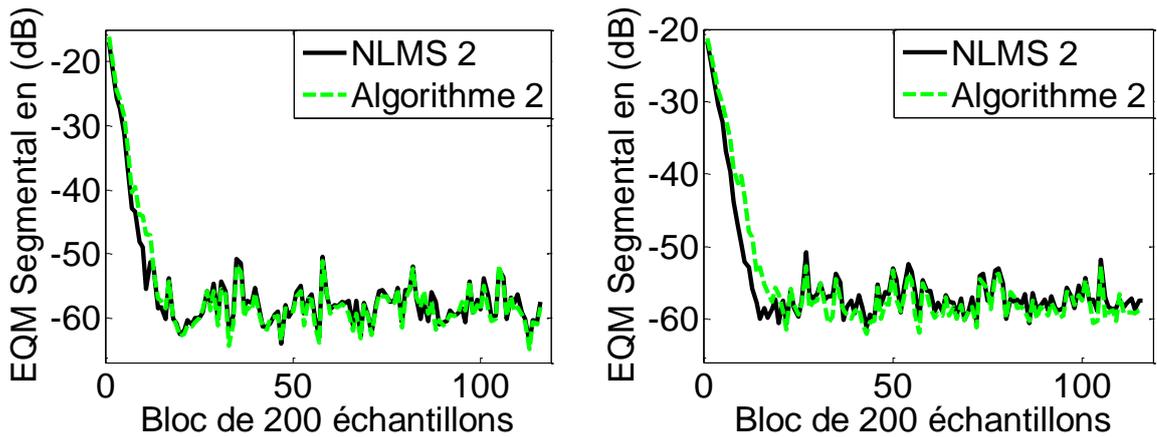


Figure 3. 25 : Evaluation du critère EQM segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

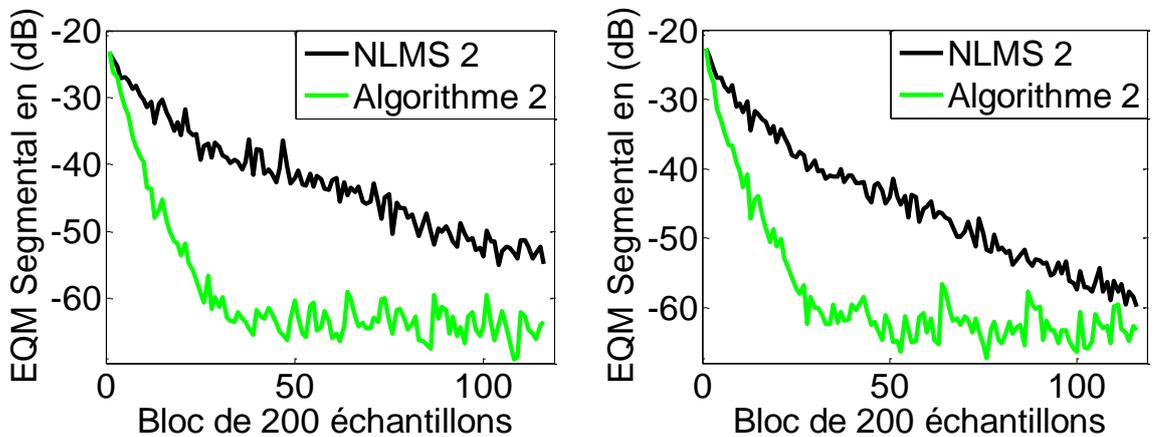


Figure 3. 26 : Evaluation du critère EQM segmental pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de la distance cepstrale*

Afin d'analyser la distorsion introduite par chacun des deux algorithmes Algorithme 2 et NLMS 2, nous mesurerons la distance cepstrale entre le signal de parole originale et celui obtenu en sortie de traitement de chaque algorithme. Les résultats obtenus pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$, et deux différents types de bruits, i.e. USASI et blanc, sont reportés sur les figures 3.27, et 3.28. Notons que les paramètres de simulation de chaque algorithme sont similaires à ceux résumés dans le tableau 3.6.

Les résultats obtenus dans le cas du bruit USASI montrent clairement que les valeurs de la DC obtenues avec l'algorithme 2 sont meilleures que celles

d'algorithme NLMS 2. Ceci est noté même lorsque le niveau de RSB d'entrée est faible. Nous avons noté également un comportement similaire en termes de la DC pour les deux algorithmes Algorithme 2 et NLMS 2 avec les composantes du bruit blanc.

Ces résultats confirment que l'algorithme 2 réduit le bruit acoustique en sortie tout en conservant l'intelligibilité du signal de parole.

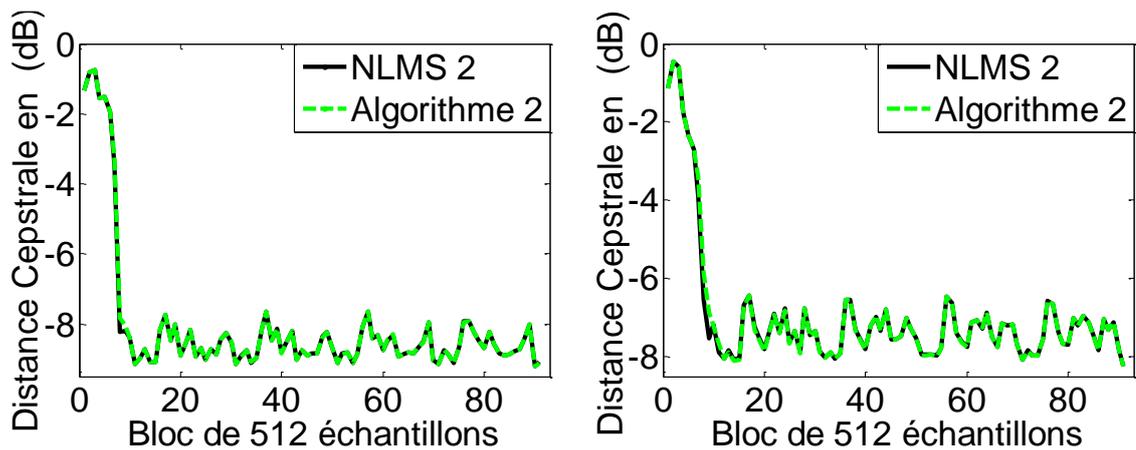


Figure 3. 27 : Evaluation du critère de la DC pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit blanc à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

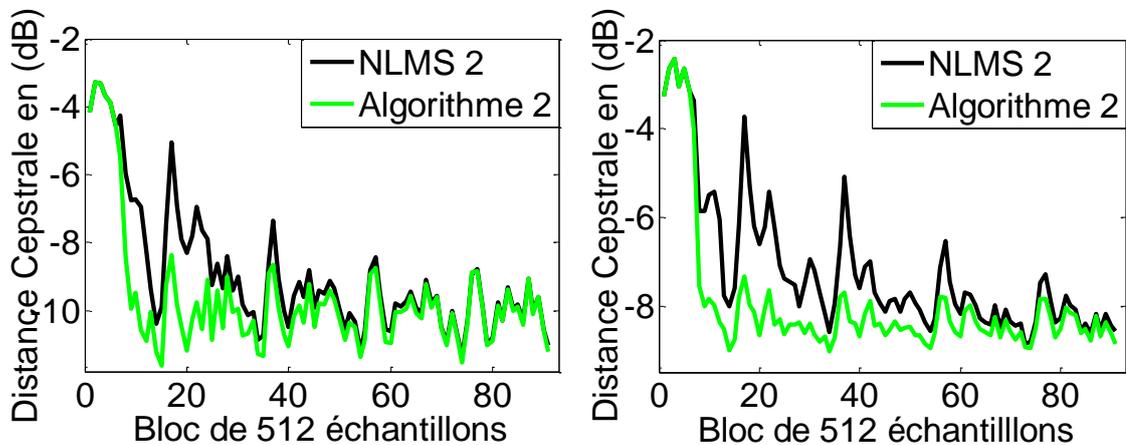


Figure 3. 28 : Evaluation du critère de la DC pour l'algorithme 2 proposé et l'algorithme NLMS 2, avec un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

3.7. Conclusion

Dans ce chapitre, nous avons présenté deux nouveaux algorithmes bicapteurs pour les applications de réduction du bruit acoustique et le rehaussement aveugle du signal de parole. Le premier algorithme, dénommé Algorithme 1, repose sur la combinaison entre la structure de séparation aveugle de sources directe et l'algorithme NLMS à convergence rapide (FNLMS). Le deuxième algorithme, dénommé Algorithme 2, est une combinaison entre la structure de séparation aveugle de source récursive et l'algorithme FNLMS. La dérivation théorique et la validation expérimentale des deux algorithmes ont été présentées le long de ce chapitre. Les deux algorithmes proposés sont numériquement stable et permettent de réduire le bruit acoustique en sortie dans des conditions aveugle, où ils ne possèdent aucune information *a priori* sur les signaux sources.

Les résultats de l'étude comparative obtenus à partir des deux algorithmes proposés (Algorithme 1 et Algorithme 2) montrent bien les bonnes performances de ces deux algorithmes en termes du rapport signal à bruit segmental, l'erreur quadratique moyenne segmental, le désajustement et la distance cepstrale, notamment lorsque le bruit utilisé est fortement corrélé.

Enfin, nous pouvons confirmer que chaque algorithme proposé constitue un bon alternatif et un bon candidat pour les applications de rehaussement du signal de parole et de la réduction du bruit acoustique.

CHAPITRE 4

NOUVEAUX ALGORITHMES BICAPTEURS EN SOUS-BANDE

4.1. Introduction

Ce chapitre présente deux nouveaux algorithmes bicapteurs en sous-bande dédiés aux applications de réduction du bruit acoustique et d'amélioration de la parole. Ces deux algorithmes sont basés sur l'implémentation en sous-bande des deux algorithmes bicapteurs présentés au chapitre 3. L'objectif derrière cette implémentation en sous-bande est d'améliorer les performances de vitesse de convergence des deux versions pleine bande.

L'analyse théorique et la validation expérimentale des deux algorithmes en sous-bande proposés seront présentées le long de ce chapitre. Les résultats d'une étude comparative entre chaque algorithme proposé et sa version pleine bande seront également présentés.

4.2. Filtrage adaptatif en sous-bande

Le filtrage adaptatif en sous-bande est devenu l'un des techniques les plus efficaces pour réduire la complexité de calcul et en même temps améliorer le taux de convergence des algorithmes dans les systèmes de traitement numérique du signal [68-71].

L'idée derrière l'implémentation en sous-bande est de décomposer un signal pleine bande en un certains nombres de sous-signaux correspondant chacun à une partie du spectre du signal original.

La structure générale du filtrage adaptatif en sous-bande est schématisée sur la figure 4.1. Les deux signaux d'entrée et désiré, i.e. $x(n)$ et $d(n)$ respectivement, sont décomposés en M sous-bandes via un banc de filtres d'analyse. Un filtrage adaptatif est ensuite appliqué pour chaque signal de sous-bande. Puisque chaque sous-bande couvre seulement une petite fraction du spectre entier du signal d'entrée, les filtres adaptatifs en sous-bande opèrent sur les signaux de sous-bandes avec des spectres plus plats par rapport au signal

plein bande, ayant pour résultat un taux de convergence sensiblement plus rapide. D'ailleurs, les filtres adaptatifs en sous-bande fonctionnent à un taux d'échantillonnage amoindri, rapportant une réduction significative de la complexité de calcul [72].

Enfin, l'erreur de filtrage $e(n)$ pleine bande est reconstruite par un banc de filtres de synthèse. Les détails de la conception des bancs de filtres d'analyse et de synthèse seront discutés dans ce qui suit.

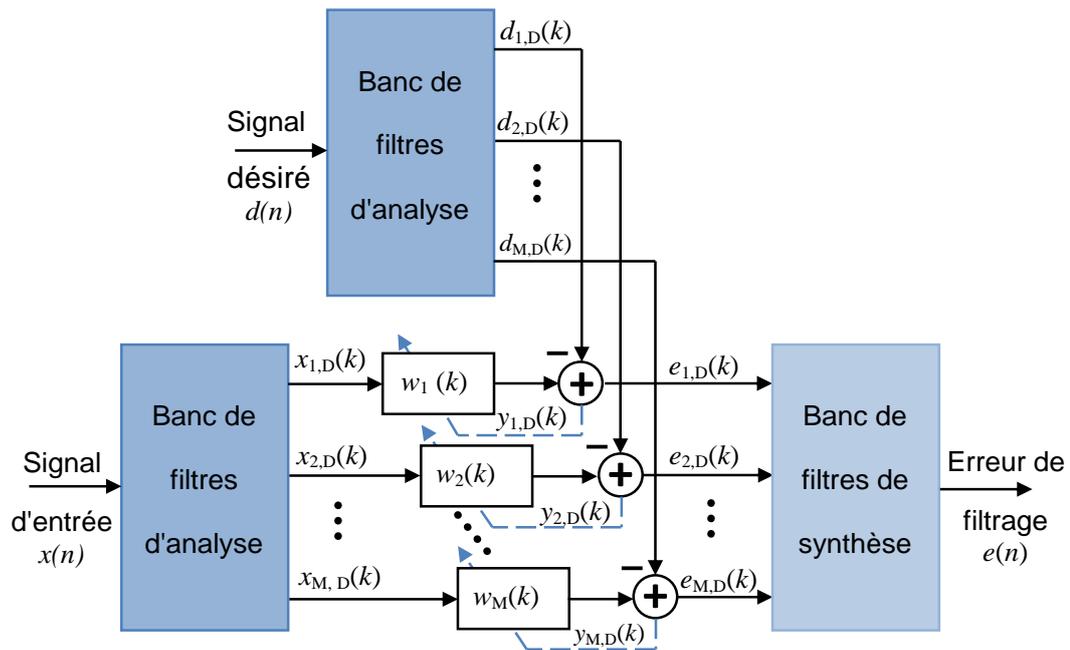


Figure 4. 1 : Structure générale du filtrage adaptatif en sous-bande.

4.2.1. Décimation et interpolation

Dans cette partie, nous décrirons brièvement le principe des opérations de décimation et d'interpolation, qui constituent des éléments essentiels pour le filtrage adaptatif en sous-bande.

- Décimation

L'opération de décimation (également appelé sous-échantillonnage) d'un facteur D appliqué à un signal numérique $x(n)$ consiste à conserver les échantillons de $x(n)$ qui se produisent à des instants de temps égaux à des multiples de D [73]. Le symbole de l'opération de décimation est représenté sur la figure 4.2.



Figure 4. 2 : Opération de décimation par un facteur D.

Le signal de sortie du décimateur peut être exprimé comme suit:

$$x_D(k) = x(kD) \quad (4.1)$$

Où, D est un facteur de décimation, n représente l'indice de temps et k représente l'indice de temps décimé.

- Interpolation

L'interpolation (où sur-échantillonnage) d'un signal numérique $x(n)$ par un facteur I implique l'insertion de $I - 1$ zéros entre chaque paire adjacente d'échantillons [73], le symbole de cette opération est représenté sur la figure 4.3.



Figure 4. 3 : Opération d'interpolation par un facteur I.

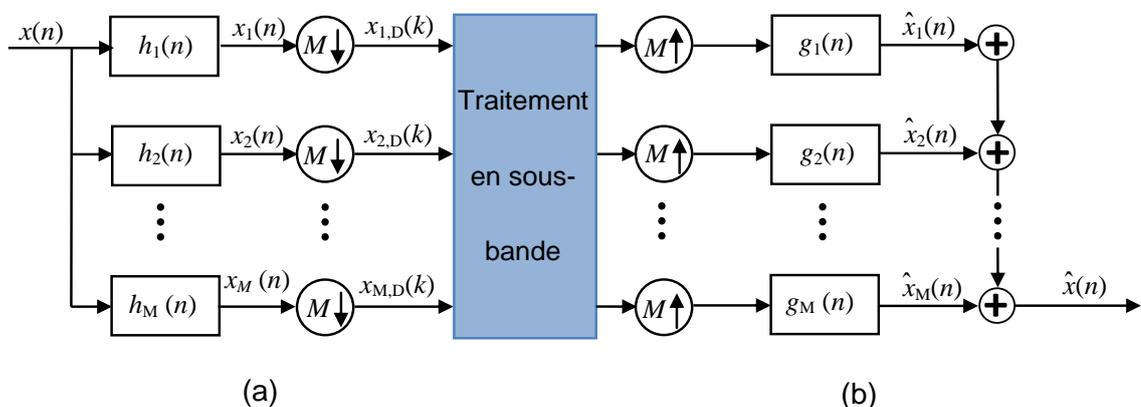
Le signal de sortie de l'interpolateur est défini par la relation suivante:

$$x_I(m) = \begin{cases} x(m/I), & m = 0, \pm I, \pm 2I, \dots \\ 0, & \text{ailleurs} \end{cases} \quad (4.2)$$

Où, I est le facteur d'interpolation, m est l'indice temporelle de la séquence interpolée $x_I(m)$.

4.2.2. Analyse et synthèse par banc de filtres

Un banc de filtres est un ensemble de filtres passe-bande avec une entrée commune pour le banc de filtres d'analyse ou une sortie sommée pour le banc de filtres de synthèse. Ce système est représenté sur la figure 4.4.

Figure 4. 4 : Banc de filtres à M canaux (a) banc d'analyse (b) banc de synthèse.

La figure 4.4 (a) représente le banc de filtres d'analyse, où les filtres h_i (avec $i = 1, 2, \dots, M$ et M est le nombre de sous-bandes) sont les filtres d'analyse suivis par des décimateurs de facteur entier ($D = M$), tandis que le banc de filtres de synthèse est représenté sur la figure 4.4 (b), où les filtres g_i sont les filtres de synthèse précédés par des interpolateurs de facteur entier ($I = M$).

Les bancs de filtres d'analyse et de synthèse sont généralement associés, le premier décompose un signal pour appliquer un traitement à chaque signal de sous-bande, et le second recombine les signaux de sous-bandes traités pour construire le signal modifié.

Un banc de filtres est appelé banc de filtres décimé de façon critique (ou maximale) si le facteur de décimation est égal au nombre de sous-bandes, i.e. $D = M$ [73]. La décimation critique préserve le taux d'échantillonnage effectif avec M signaux de sous-bandes décimés $x_{i,D}(k)$ ($i = 1, 2, \dots, M$), chacun avec $1/M$ de la fréquence d'échantillonnage d'origine, de sorte que le nombre total d'échantillons de sous-bandes est identique à celui du signal pleine bande $x(n)$. Dans la section de synthèse, les signaux de sous-bandes décimés sont interpolés par le même facteur avant d'être recombinaés par le banc de filtres de synthèse. Par conséquent, la fréquence d'échantillonnage d'origine est restaurée dans le signal reconstruit en pleine bande.

4.3. Algorithmes bicapteurs en sous-bande basés sur la BSS

Dans cette section, nous allons présenter deux algorithmes bicapteurs en sous-bande appelés: Algorithme 3 et Algorithme 4, pour les applications de réduction du bruit acoustique et le rehaussement du signal de parole dans les systèmes de communication mains-libres.

Ces deux algorithmes (Algorithme 3 et Algorithme 4) sont basés, respectivement, sur l'implémentation en sous-bande de l'Algorithme 1 et l'Algorithme 2 présentés au chapitre 3. L'intérêt de cette implémentation en sous-bande suggérée est d'améliorer les performances des deux algorithmes Algorithme 1 et Algorithme 2 en termes de vitesse de convergence.

4.3.1. Algorithme 3

Cet algorithme est une implémentation en sous-bande du premier algorithme proposé au chapitre 3, qui correspond à une combinaison entre la structure de

séparation aveugle de sources directe (FBSS) et l'algorithme FNLMS [74]. La figure 4.5 présente le schéma détaillé de l'algorithme proposé Algorithme 3. La formulation mathématique de l'algorithme proposé se fait en plusieurs étapes:

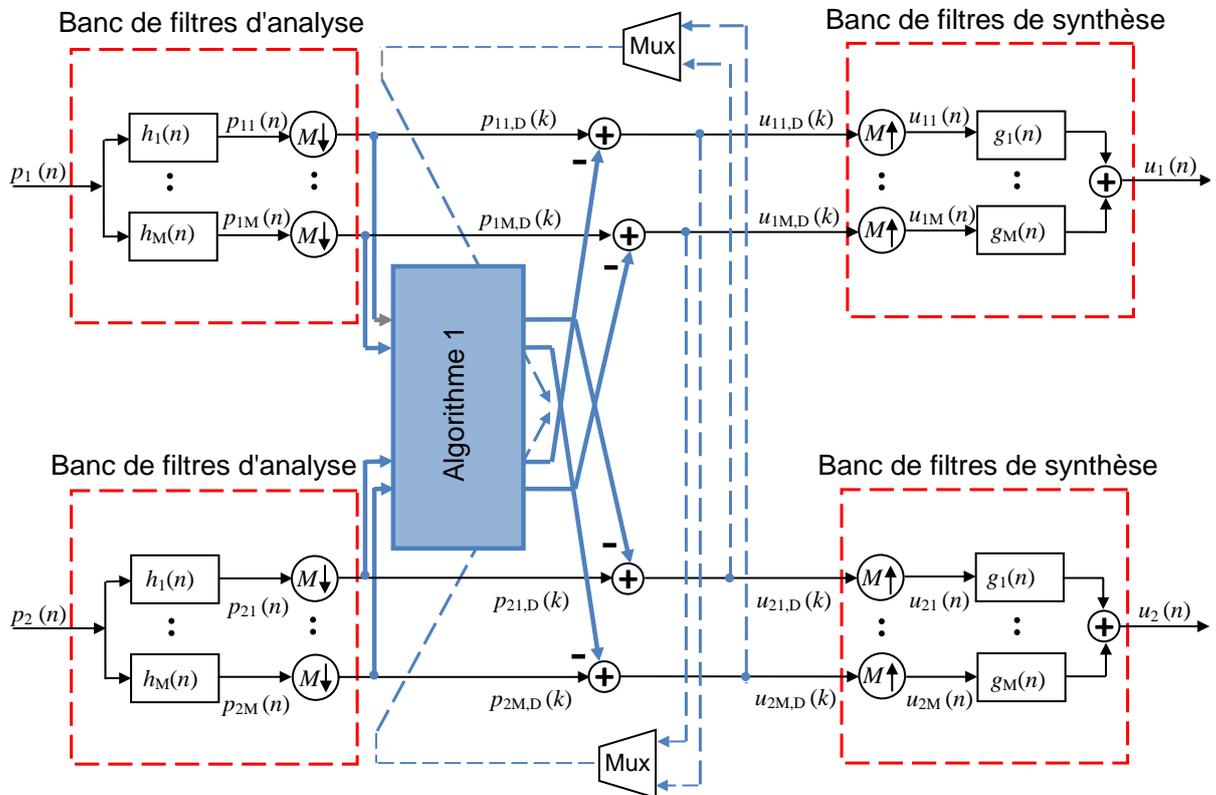


Figure 4. 5 : Schéma détaillé de l'Algorithme 3.

– Etape 1: dans la première étape, un banc de filtres d'analyse $h_1(n), h_2(n), \dots, h_M(n)$ est utilisé pour décomposer les signaux d'observations pleine bande $p_1(n)$ et $p_2(n)$ en M signaux de sous-bandes $p_{1i}(n)$ et $p_{2i}(n)$ (avec $i = 1, 2, \dots, M$). Ces signaux de sous-bandes résultant sont ensuite décimés par un facteur de décimation D égal au nombre de sous-bandes M , i.e. $M = D$. Les sous-signaux décimés observés sont définis par:

$$p_{1i,D}(k) = p_{1i}(kM) \quad i = 1, 2, \dots, M. \quad (4.3)$$

$$p_{1i}(n) = \mathbf{h}_i^T(n) \mathbf{p}_1(n) \quad i = 1, 2, \dots, M. \quad (4.4)$$

$$p_{2i,D}(k) = p_{2i}(kM) \quad i = 1, 2, \dots, M. \quad (4.5)$$

$$p_{2i}(n) = \mathbf{h}_i^T(n) \mathbf{p}_2(n) \quad i = 1, 2, \dots, M. \quad (4.6)$$

Où, M est le nombre de sous-bandes, k est l'indice de temps des signaux de sous-bandes décimés. $\mathbf{p}_1(n) = [p_1(n), p_1(n-1), \dots, p_1(n-l+1)]$ et $\mathbf{p}_2(n) =$

$p_2(n), p_2(n-1), \dots, p_2(n-l+1)$ sont les vecteurs des signaux d'observations $p_1(n)$ et $p_2(n)$. l est la longueur des filtres d'analyse $h_i(n)$.

– Etape 2: dans la deuxième étape, nous appliquons la structure de séparation aveugle de sources directe pour récupérer les signaux sources $s(n)$ et $b(n)$ à partir seulement des sous-signaux décimés observés $p_{1i,D}(k)$ et $p_{2i,D}(k)$. Cette structure a besoin de deux filtres adaptatifs pour extraire le signal de parole originale du bruit. Pour mettre à jour les coefficients de ces filtres adaptatifs nous utilisons l'Algorithme1.

Dans l'Algorithme 3 proposé, les signaux de sortie sont estimés en sous bandes, tandis que les coefficients des deux filtres adaptatifs sont explicitement adaptés dans leurs formes pleine bande. Ce mécanisme d'adaptation est différent de celui de la structure conventionnelle du filtrage adaptatif en sous-bande, où chaque sous-bande a son propre sous-filtre et boucle d'adaptation [75-76]. Le processus d'adaptation en sous-bande de l'algorithme proposé est schématisé sur la figure 4.6.

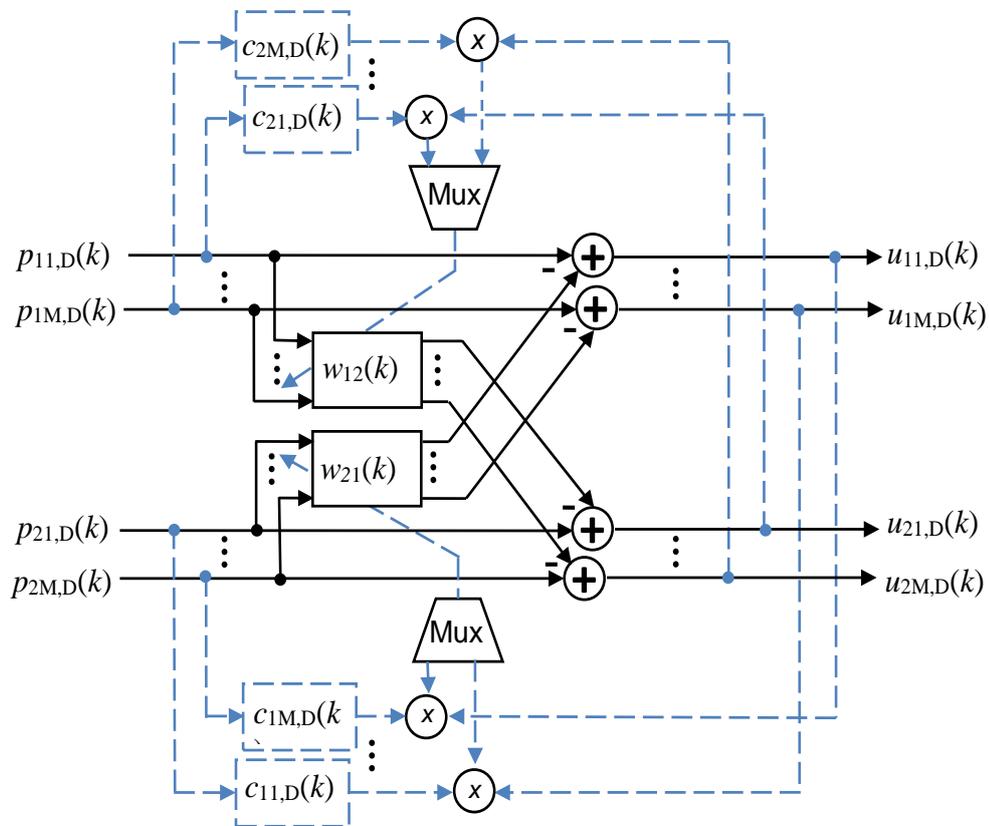


Figure 4. 6 : Schéma détaillé du processus d'adaptation de l'Algorithme 3.

Les sous-signaux estimés $u_{1i,D}(k)$ et $u_{2i,D}(k)$ sont donnés comme suit:

$$u_{1i,D}(k) = p_{1i,D}(k) - \mathbf{w}_{21}^T(k-1)\mathbf{p}_{2i,D}(k) \quad i = 1, 2, \dots, M. \quad (4.7)$$

$$u_{2i,D}(k) = p_{2i,D}(k) - \mathbf{w}_{12}^T(k-1)\mathbf{p}_{1i,D}(k) \quad i = 1, 2, \dots, M. \quad (4.8)$$

Où, $\mathbf{p}_{1i,D}(k) = [p_{1i,D}(k), p_{1i,D}(k-1), \dots, p_{1i,D}(k-L+1)]$ et $\mathbf{p}_{2i,D}(k) = [p_{2i,D}(k), p_{2i,D}(k-1), \dots, p_{2i,D}(k-L+1)]$ sont respectivement, les vecteurs des sous-signaux décimés $p_{1i,D}(k)$ et $p_{2i,D}(k)$. $\mathbf{w}_{21}(k) = [w_{21}(k), w_{21}(k-1), \dots, w_{21}(k-L+1)]^T$ et $\mathbf{w}_{12}(k) = [w_{12}(k), w_{12}(k-1), \dots, w_{12}(k-L+1)]^T$ sont les vecteurs des coefficients des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$, respectivement. L est la longueur de ces deux filtres adaptatifs.

Les équations de mise à jour des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$, sont données par les deux formules suivantes :

$$\mathbf{w}_{21}(k) = \mathbf{w}_{21}(k-1) - \mu_1 \sum_{i=1}^M [u_{1i,D}(k)\mathbf{c}_{1i,D}(k)] \quad (4.9)$$

$$\mathbf{w}_{12}(k) = \mathbf{w}_{12}(k-1) - \mu_2 \sum_{i=1}^M [u_{2i,D}(k)\mathbf{c}_{2i,D}(k)] \quad (4.10)$$

Où, μ_1 et μ_2 ($0 < \mu_1, \mu_2 < 2$) sont respectivement les pas d'adaptation des deux filtres adaptatifs $\mathbf{w}_{21}(k)$ et $\mathbf{w}_{12}(k)$.

$\mathbf{c}_{1i,D}(k)$ et $\mathbf{c}_{2i,D}(k)$ sont les vecteurs de gain d'adaptation de sous-bandes décimés, qui sont donnés par les relations suivantes:

$$\mathbf{c}_{1i,D}(k) = \gamma_{1i,D}(k)\check{\mathbf{k}}_{1i,D}(k) \quad (4.11)$$

$$\mathbf{c}_{2i,D}(k) = \gamma_{2i,D}(k)\check{\mathbf{k}}_{2i,D}(k) \quad (4.12)$$

Les scalaires $\gamma_{1i,D}(k)$ et $\gamma_{2i,D}(k)$ sont les variables de vraisemblance de sous-bandes décimées, définies comme suit:

$$\gamma_{1i,D}(k) = \frac{1}{1 - \check{\mathbf{k}}_{1i,D}^T(k)\mathbf{p}_{2i,D}(k)} \quad (4.13)$$

$$\gamma_{2i,D}(k) = \frac{1}{1 - \check{\mathbf{k}}_{2i,D}^T(k)\mathbf{p}_{1i,D}(k)} \quad (4.14)$$

Les vecteurs de sous-bandes décimés $\check{\mathbf{k}}_{1i,D}(k)$ et $\check{\mathbf{k}}_{2i,D}(k)$ sont les gains de Kalman duaux, qui sont donnés par les relations suivantes:

$$\begin{bmatrix} \check{\mathbf{k}}_{1i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_{1i,D}(k)}{\lambda_f \alpha_{1i,D}(k-1) + c_0} \\ \check{\mathbf{k}}_{1i,D}(k-1) \end{bmatrix} \quad (4.15)$$

$$\begin{bmatrix} \check{\mathbf{k}}_{2i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_{2i,D}(k)}{\lambda_f \alpha_{2i,D}(k-1) + c_0} \\ \check{\mathbf{k}}_{2i,D}(k-1) \end{bmatrix} \quad (4.16)$$

Le paramètre λ_f ($0 < \lambda_f < 1$) est un facteur d'oubli exponentiel, c_0 est un petit scalaire positif utilisé pour éviter la division par de petites valeurs dans les périodes d'inactivité du signal vocal (périodes de silence) et l'astérisque (*) représente le dernier élément inutile des vecteurs de sous-bandes décimés $\check{\mathbf{k}}_{1i,D}(k)$ et $\check{\mathbf{k}}_{2i,D}(k)$. Les paramètres de sous-bandes décimés $\alpha_{1i,D}(k)$ et $\alpha_{2i,D}(k)$ sont les variances des erreurs de prédiction aller, elles sont estimées récursivement par les relations suivantes :

$$\alpha_{1i,D}(k) = \lambda_f \alpha_{1i,D}(k-1) + e_{1i,D}^2(k) \quad (4.17)$$

$$\alpha_{2i,D}(k) = \lambda_f \alpha_{2i,D}(k-1) + e_{2i,D}^2(k) \quad (4.18)$$

Les erreurs de prédiction aller de sous-bandes décimées $e_{1i,D}(k)$ et $e_{2i,D}(k)$ peuvent être calculées par les relations suivantes:

$$e_{1i,D}(k) = p_{2i,D}(k) - a_{1i,D} p_{2i,D}(k-1) \quad (4.19)$$

$$e_{2i,D}(k) = p_{1i,D}(k) - a_{2i,D} p_{1i,D}(k-1) \quad (4.20)$$

Où, $a_{1i,D}$ et $a_{2i,D}$ sont les coefficients de prédiction de sous-bandes décimés, ils sont obtenus en minimisant, respectivement à chaque sous-bande les fonctions $E[e_{1i,D}(k)]$ et $E[e_{2i,D}(k)]$:

$$a_{1i,D} = \frac{E[p_{2i,D}(k)p_{2i,D}(k-1)]}{E[p_{2i,D}^2(k-1)]} = \frac{r_{1i,D}}{r_{2i,D}} \quad (4.21)$$

$$a_{2i,D} = \frac{E[p_{1i,D}(k)p_{1i,D}(k-1)]}{E[p_{1i,D}^2(k-1)]} = \frac{r_{3i,D}}{r_{4i,D}} \quad (4.22)$$

Où, $r_{1i,D}$ et $r_{3i,D}$ désignent respectivement, le premier coefficient de la fonction d'auto-corrélation des sous-signaux décimés observés $p_{2i,D}(k)$ et $p_{1i,D}(k)$. r_2 et r_4 désignent respectivement, la puissance des sous-signaux décimés observés $p_{2i,D}(k)$ et $p_{1i,D}(k)$.

Ces dernières relations sont évaluées comme dans l'Algorithme 1 par des estimateurs récursifs:

$$a_{1i,D}(k) = \frac{r_{1i,D}(k)}{r_{2i,D}(k) + c_f} \quad (4.23)$$

$$a_{2i,D}(k) = \frac{r_{3i,D}(k)}{r_{4i,D}(k) + c_f} \quad (4.24)$$

Avec:

$$r_{1i,D}(k) = \eta_f r_{1i,D}(k-1) + p_{2i,D}(k) p_{2i,D}(k-1) \quad (4.25)$$

$$r_{2i,D}(k) = \eta_f r_{2i,D}(k-1) + p_{2i,D}^2(k) \quad (4.26)$$

$$r_{3i,D}(k) = \eta_f r_{3i,D}(k-1) + p_{1i,D}(k) p_{1i,D}(k-1) \quad (4.27)$$

$$r_{4i,D}(k) = \eta_f r_{4i,D}(k-1) + p_{1i,D}^2(k) \quad (4.28)$$

Où, η_f ($0 < \eta_f < 1$) est un facteur d'oubli, et c_f est une constante de régularisation.

– Etape 3: Dans cette dernière étape, les signaux estimés $u_1(n)$ et $u_2(n)$ pleine bande sont reconstruits en interpolant et en recombinaison les M sous-signaux estimés $u_{1i,D}(k)$ et $u_{2i,D}(k)$ à l'aide d'un banc de filtres de synthèse $g_1(n), g_2(n), \dots, g_M(n)$.

Les signaux estimés $u_1(n)$ et $u_2(n)$ pleine bande peuvent être exprimés comme suit:

$$u_1(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{U}_{1i}(n) \quad (4.29)$$

$$u_2(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{U}_{2i}(n) \quad (4.30)$$

Où, $\mathbf{U}_{1i}(n) = [u_{1i}(n), u_{1i}(n-1), \dots, \dots, u_{1i}(n-l+1)]$, $\mathbf{U}_{2i}(n) = [u_{2i}(n), u_{2i}(n-1), \dots, \dots, u_{2i}(n-l+1)]$ sont respectivement, les vecteurs des signaux estimés de sous-bandes $u_{1i}(n)$ et $u_{2i}(n)$.

Avec:

$$u_{1i}(n) = \begin{cases} u_{1i,D}(n/I) & , \quad n = 0, \pm I, \pm 2I, \dots \\ 0 & , \quad \text{ailleurs} \end{cases} \quad (4.31)$$

$$u_{2i}(n) = \begin{cases} u_{2i,D}(n/I) & , \quad n = 0, \pm I, \pm 2I, \dots \\ 0 & , \quad \text{ailleurs} \end{cases} \quad (4.32)$$

I est un facteur d'interpolation, ce facteur est pris égal au nombre de sous-bandes M .

Le tableau 4.1 résume les équations de fonctionnement de l'Algorithme 3.

Tableau 4. 1 : Résumé de l'Algorithme 3

Initialisation

$$\mathbf{w}_{21}(0) = \mathbf{w}_{12}(0) = 0, \mathbf{c}_{1i,D}(0) = \mathbf{c}_{2i,D}(0) = 0, \alpha_{1i,D}(0) = \alpha_{2i,D}(0) = E_0, r_{1i,D}(0) = r_{3i,D}(0) = 0, r_{2i,D}(0) = r_{4i,D}(0) = E_0$$

Décomposition en sous-bandes

$$p_{1i}(n) = \mathbf{h}_i^T(n) \mathbf{p}_1(n), p_{2i}(n) = \mathbf{h}_i^T(n) \mathbf{p}_2(n), \text{ avec } i = 1, \dots, M,$$

Prédiction

Erreurs de prédiction de sous-bandes:

$$e_{1i,D}(k) = p_{2i,D}(k) - a_{1i,D} p_{2i,D}(k-1), e_{2i,D}(k) = p_{1i,D}(k) - a_{2i,D} p_{1i,D}(k-1)$$

Coefficients de prédiction de sous-bande:

$$a_{1i,D}(k) = \frac{r_{1i,D}(k)}{r_{2i,D}(k) + c_f}, a_{2i,D}(k) = \frac{r_{3i,D}(k)}{r_{4i,D}(k) + c_f}$$

$$r_{1i,D}(k) = \eta_f r_{1i,D}(k-1) + p_{2i,D}(k) p_{2i,D}(k-1), r_{2i,D}(k) = \eta_f r_{2i,D}(k-1) + p_{2i,D}^2(k)$$

$$r_{3i,D}(k) = \eta_f r_{3i,D}(k-1) + p_{1i,D}(k) p_{1i,D}(k-1), r_{4i,D}(k) = \eta_f r_{4i,D}(k-1) + p_{1i,D}^2(k)$$

Variances des erreurs de prédiction aller de sous-bandes:

$$\alpha_{1i,D}(k) = \lambda_f \alpha_{1i,D}(k-1) + e_{1i,D}^2(k), \alpha_{2i,D}(k) = \lambda_f \alpha_{2i,D}(k-1) + e_{2i,D}^2(k)$$

Vecteurs gains de Kalman duaux de sous-bandes:

$$\begin{bmatrix} \check{\mathbf{k}}_{1i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_{1i,D}(k)}{\lambda_f \alpha_{1i,D}(k-1) + c_0} \\ \check{\mathbf{k}}_{1i,D}(k-1) \end{bmatrix}, \begin{bmatrix} \check{\mathbf{k}}_{2i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{e_{2i,D}(k)}{\lambda_f \alpha_{2i,D}(k-1) + c_0} \\ \check{\mathbf{k}}_{2i,D}(k-1) \end{bmatrix}$$

Variables de vraisemblance de sous-bande:

$$\gamma_{1i,D}(k) = \frac{1}{1 - \check{\mathbf{k}}_{1i,D}^T(k) \mathbf{p}_{2i,D}(k)}, \gamma_{2i,D}(k) = \frac{1}{1 - \check{\mathbf{k}}_{2i,D}^T(k) \mathbf{p}_{1i,D}(k)}$$

Vecteurs gains d'adaptation de sous-bandes:

$$\mathbf{c}_{1i,D}(k) = \gamma_{1i,D}(k) \check{\mathbf{k}}_{1i,D}(k), \mathbf{c}_{2i,D}(k) = \gamma_{2i,D}(k) \check{\mathbf{k}}_{2i,D}(k)$$

Estimation des sous-signaux de sortie

$$u_{1i,D}(k) = p_{1i,D}(k) - \mathbf{w}_{21}^T(k-1) \mathbf{p}_{2i,D}(k), u_{2i,D}(k) = p_{2i,D}(k) - \mathbf{w}_{12}^T(k-1) \mathbf{p}_{1i,D}(k)$$

Adaptation

Equations de mise à jour des filtres adaptatifs:

$$\mathbf{w}_{21}(k) = \mathbf{w}_{21}(k-1) - \mu_1 \sum_{i=1}^M [u_{1i,D}(k) \mathbf{c}_{1i,D}(k)], \mathbf{w}_{12}(k) = \mathbf{w}_{12}(k-1) - \mu_2 \sum_{i=1}^M [u_{2i,D}(k) \mathbf{c}_{2i,D}(k)],$$

Synthèse

Signaux de sortie pleine bande

$$u_1(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{U}_{1i}(n), u_2(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{U}_{2i}(n)$$

NB: La réalisation de cet algorithme est effectuée sans décimation.

4.3.2. Algorithme 4

La figure 4.7 présente le schéma détaillé de l'Algorithme 4 proposé. Cet algorithme est une implémentation en sous-bande de l'Algorithme 2 présenté au chapitre 3, c'est à dire que nous avons implémenté en sous-bande l'algorithme FNLMS combiné avec la structure de séparation aveugle de sources récursive [77].

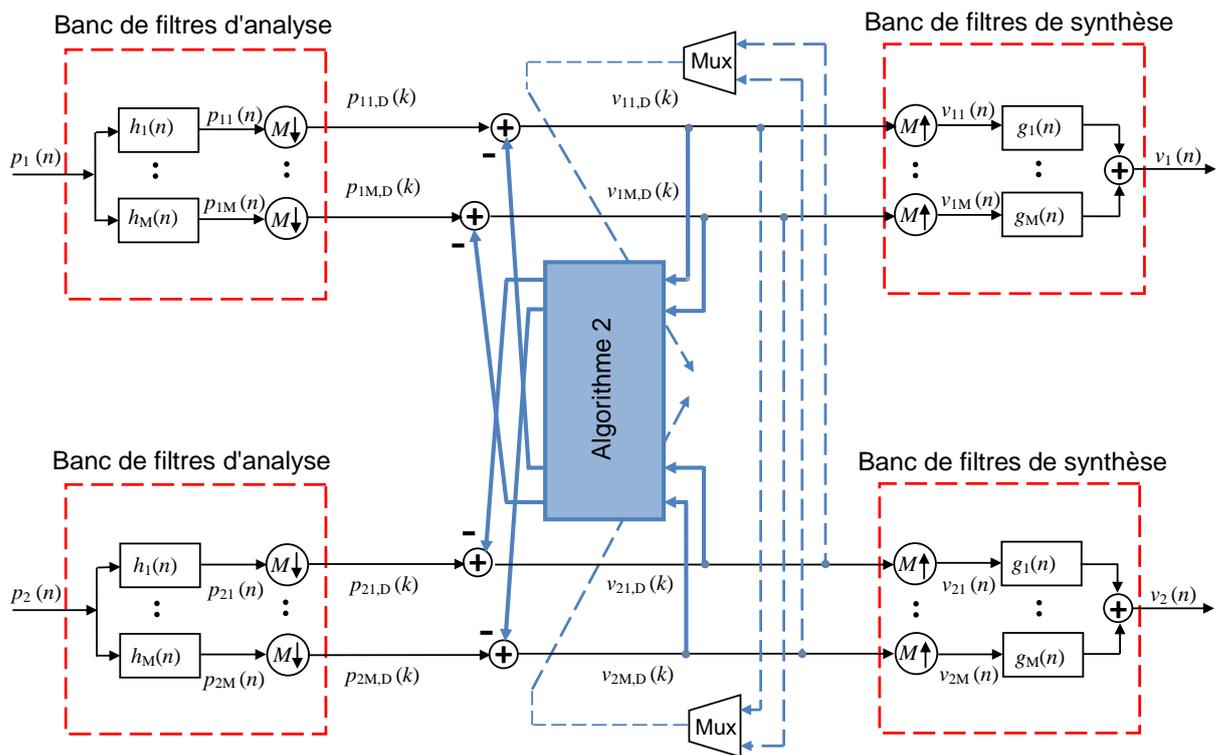


Figure 4. 7 : Schéma détaillé de l'Algorithme 4.

La dérivation mathématique de l'Algorithme 4 est donnée par les étapes suivantes:

- Etape 1: dans la première étape, nous avons repris la même démarche de décomposition en sous-bandes que celle de l'Algorithme 3, où les signaux d'observations pleine bande $p_1(n)$ et $p_2(n)$ sont partitionnés en M sous-bandes au moyen de filtres d'analyse $h_1(n), h_2(n), \dots, h_M(n)$. Les signaux de sortie de chaque sous-bande $p_{1i}(n)$ et $p_{2i}(n)$ avec $i = 1, 2, \dots, M$ sont ensuite décimés par un facteur $D = M$, ce qui génère les nouveaux sous-signaux décimés $p_{1i,D}(k)$ et $p_{2i,D}(k)$, (où k est l'indice de temps décimé).

- Etape 2: après l'étape de partitionnement en sous-bandes et de décimation des signaux d'observations pleine bande, la structure de séparation aveugle de sources récursive est appliquée pour estimer les deux ensembles des sous-

signaux de parole $v_{1i,D}(k)$ et de bruit $v_{2i,D}(k)$ à partir seulement des deux ensembles des sous-signaux décimés observés $p_{1i,D}(k)$ et $p_{2i,D}(k)$.

Les sous-signaux estimés $v_{1i,D}(k)$ et $v_{2i,D}(k)$ sont donnés par les formules suivantes:

$$v_{1i,D}(k) = p_{1i,D}(k) - \mathbf{w}_{21}^T(k-1)\mathbf{v}_{2i,D}(k) \quad i = 1, 2, \dots, M. \quad (4.33)$$

$$v_{2i,D}(k) = p_{2i,D}(k) - \mathbf{w}_{12}^T(k-1)\mathbf{v}_{1i,D}(k) \quad i = 1, 2, \dots, M. \quad (4.34)$$

Où, $p_{1i,D}(k)$ et $p_{2i,D}(k)$ sont les sous-signaux décimés observés donnés respectivement par les relations (4.3) et (4.5). $\mathbf{v}_{1i,D}(k) = [\mathbf{v}_{1i,D}(k), \mathbf{v}_{1i,D}(k-1), \dots, \mathbf{v}_{1i,D}(k-L+1)]^T$ et $\mathbf{v}_{2i,D}(k) = [\mathbf{v}_{2i,D}(k), \mathbf{v}_{2i,D}(k-1), \dots, \mathbf{v}_{2i,D}(k-L+1)]^T$ sont respectivement, les vecteurs des sous-signaux de sortie décimés $v_{1i,D}(k)$ et $v_{2i,D}(k)$. $\mathbf{w}_{21}(k) = [w_{21}(k), w_{21}(k-1), \dots, w_{21}(k-L+1)]^T$ et $\mathbf{w}_{12}(k) = [w_{12}(k), w_{12}(k-1), \dots, w_{12}(k-L+1)]^T$ sont respectivement, les vecteurs des coefficients des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$. L est la longueur de ces deux filtres adaptatifs.

Les coefficients des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$ sont adaptés récursivement par l'algorithme 2. Le schéma détaillé du processus d'adaptation est donné par la figure 4.8.

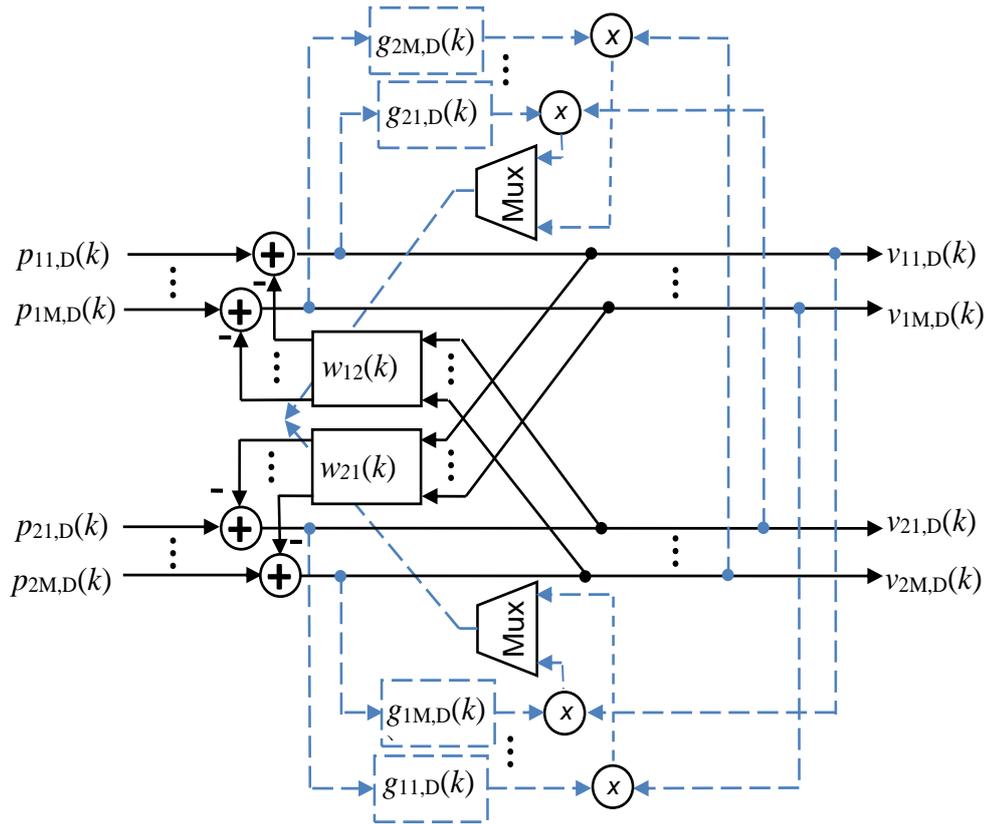


Figure 4. 8 : Schéma détaillé du processus d'adaptation de l'Algorithme 4.

A partir de la figure 4.8, nous pouvons déduire que les relations de mise à jour des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$ peuvent être exprimées comme suit:

$$\mathbf{w}_{21}(k) = \mathbf{w}_{21}(k-1) - \mu_{21} \sum_{i=1}^M [v_{1i,D}(k) \mathbf{g}_{1i,D}(k)] \quad (4.35)$$

$$\mathbf{w}_{12}(k) = \mathbf{w}_{12}(k-1) - \mu_{12} \sum_{i=1}^M [v_{2i,D}(k) \mathbf{g}_{2i,D}(k)] \quad (4.36)$$

Où, μ_{21} et μ_{12} ($0 < \mu_{21}, \mu_{12} < 2$) sont respectivement les pas d'adaptation qui assurent la convergence et la stabilité des deux filtres adaptatifs $w_{21}(k)$ et $w_{12}(k)$.

$\mathbf{g}_{1i,D}(k)$ et $\mathbf{g}_{2i,D}(k)$ sont les vecteurs de gain d'adaptation de sous-bandes décimés, définis par :

$$\mathbf{g}_{1i,D}(k) = \Gamma_{1i,D}(k) \tilde{\mathbf{k}}_{1i,D}(k) \quad (4.37)$$

$$\mathbf{g}_{2i,D}(k) = \Gamma_{2i,D}(k) \tilde{\mathbf{k}}_{2i,D}(k) \quad (4.38)$$

Où, $\Gamma_{1i,D}(k)$, $\Gamma_{2i,D}(k)$ et $\tilde{\mathbf{k}}_{1i,D}(k)$, $\tilde{\mathbf{k}}_{2i,D}(k)$ désignent respectivement les variables de vraisemblance de sous-bandes décimés et les vecteurs gains de Kalman duaux de sous-bandes décimés.

Les variables de vraisemblance de sous-bandes décimés sont définies par:

$$\Gamma_{1i,D}(k) = \frac{1}{1 - \tilde{\mathbf{k}}_{1i,D}^T(k) \mathbf{v}_{2i,D}(k)} \quad (4.39)$$

$$\Gamma_{2i,D}(k) = \frac{1}{1 - \tilde{\mathbf{k}}_{2i,D}^T(k) \mathbf{v}_{1i,D}(k)} \quad (4.40)$$

Les relations de mise à jour des deux vecteurs $\tilde{\mathbf{k}}_{1i,D}(k)$ et $\tilde{\mathbf{k}}_{2i,D}(k)$ à chaque sous-bande sont données comme suit:

$$\begin{bmatrix} \tilde{\mathbf{k}}_{1i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_{1i,D}(k)}{\lambda_b a_{1i,D}(k-1) + \xi} \\ \tilde{\mathbf{k}}_{1i,D}(k-1) \end{bmatrix} \quad (4.41)$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_{2i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_{2i,D}(k)}{\lambda_b a_{2i,D}(k-1) + \xi} \\ \tilde{\mathbf{k}}_{2i,D}(k-1) \end{bmatrix} \quad (4.42)$$

Où, le paramètre λ_b est un facteur d'oubli exponentiel qui doit satisfaire la condition : $0 < \lambda_b < 1$, ξ est une petite constante positive utilisée pour éviter la division par de petites valeurs dans les périodes de silence. L'astérisque (*) représente le dernier élément inutile des vecteurs de sous-bandes décimés $\tilde{\mathbf{k}}_{1i,D}(k)$ et $\tilde{\mathbf{k}}_{2i,D}(k)$. Les paramètres de sous-bande décimés $a_{1i,D}(k)$ et $a_{2i,D}(k)$ sont les variances des erreurs de prédiction aller. Elles sont estimées à chaque sous-bande comme suit:

$$a_{1i,D}(k) = \lambda_b a_{1i,D}(k-1) + \varepsilon_{1i,D}^2(k) \quad (4.43)$$

$$a_{2i,D}(k) = \lambda_b a_{2i,D}(k-1) + \varepsilon_{2i,D}^2(k) \quad (4.44)$$

$\varepsilon_{1i,D}(k)$ et $\varepsilon_{2i,D}(k)$ sont les erreurs de prédiction aller de sous-bandes décimées, elles sont calculées à chaque sous-bande comme suit:

$$\varepsilon_{1i,D}(k) = v_{2i,D}(k) - A_{1i,D} v_{2i,D}(k-1) \quad (4.45)$$

$$\varepsilon_{2i,D}(k) = v_{1i,D}(k) - A_{2i,D} v_{1i,D}(k-1) \quad (4.46)$$

Où, $A_{1i,D}$ et $A_{2i,D}$ sont les coefficients de prédiction de sous-bandes décimés, et sont obtenus en minimisant, respectivement à chaque sous-bande les fonctions $E[\varepsilon_{1i,D}(k)]$ et $E[\varepsilon_{2i,D}(k)]$:

$$A_{1i,D} = \frac{E[v_{2i,D}(k)v_{2i,D}(k-1)]}{E[v_{2i,D}^2(k-1)]} = \frac{R_{1i,D}}{R_{2i,D}} \quad (4.47)$$

$$A_{2i,D} = \frac{E[v_{1i,D}(k)v_{1i,D}(k-1)]}{E[v_{1i,D}^2(k-1)]} = \frac{R_{3i,D}}{R_{4i,D}} \quad (4.48)$$

Où, $R_{1i,D}$ et $R_{3i,D}$ désignent respectivement, le premier coefficient de la fonction d'auto-corrélation des sous-signaux de sortie $v_{2i,D}(k)$ et $v_{1i,D}(k)$. $R_{2i,D}$ et $R_{4i,D}$ désignent respectivement, la puissance des sous-signaux de sortie $v_{2i,D}(k)$ et $v_{1i,D}(k)$.

Les relations (4.47) et (4.48) sont évaluées par des estimateurs récursifs à chaque sous-bande comme suit:

$$A_{1i,D}(k) = \frac{R_{1i,D}(k)}{R_{2i,D}(k) + c_b} \quad (4.49)$$

$$A_{2i,D}(k) = \frac{R_{3i,D}(k)}{R_{4i,D}(k) + c_b} \quad (4.50)$$

Avec:

$$R_{1i,D}(k) = \eta_b R_{1i,D}(k-1) + v_{2i,D}(k) v_{2i,D}(k-1) \quad (4.51)$$

$$R_{2i,D}(k) = \eta_b R_{2i,D}(k-1) + v_{2i,D}^2(k) \quad (4.52)$$

$$R_{3i,D}(k) = \eta_b R_{3i,D}(k-1) + v_{1i,D}(k) v_{1i,D}(k-1) \quad (4.53)$$

$$R_{4i,D}(k) = \eta_b R_{4i,D}(k-1) + v_{1i,D}^2(k) \quad (4.54)$$

Où, η_b ($0 < \eta_b < 1$) est un facteur d'oubli, et c_b est une constante de régularisation.

– Etape 3: Afin de reconstruire et synthétiser les signaux de sortie pleine bande $v_1(n)$ et $v_2(n)$, les sous-signaux estimés $v_{1i,D}(k)$ et $v_{2i,D}(k)$ sont interpolés par un facteur I égal au nombre de sous-bandes M , i.e. $I = M$. Ces derniers signaux interpolés sont ensuite filtrés par des filtres de synthèse $g_1(n), g_2(n), \dots, g_M(n)$, pour former les signaux de sortie pleine bande qui sont donnés par les relations suivantes:

$$v_1(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{V}_{1i}(n) \quad (4.55)$$

$$v_2(n) = \sum_{i=1}^M \mathbf{g}_i^T(n) \mathbf{V}_{2i}(n) \quad (4.56)$$

Où

$$v_{1i}(n) = \begin{cases} v_{1i,D}(n/M) & , \quad n = 0, \pm M, \pm 2M, \dots \\ 0 & , \quad \text{ailleurs} \end{cases} \quad (4.57)$$

$$v_{2i}(n) = \begin{cases} v_{2i,D}(n/M) & , \quad n = 0, \pm M, \pm 2M, \dots \\ 0 & , \quad \text{ailleurs} \end{cases} \quad (4.58)$$

et $\mathbf{V}_{1i}(n) = [v_{1i}(n), v_{1i}(n-1), \dots, v_{1i}(n-l+1)]$, $\mathbf{V}_{2i}(n) = [v_{2i}(n), v_{2i}(n-1), \dots, v_{2i}(n-l+1)]$. l est la longueur des filtres de synthèse.

Le tableau 4.2 résume les équations de fonctionnement de l'Algorithme 4.

Tableau 4. 2 : Résumé de l'Algorithme 4.

Initialisation
$\mathbf{v}_{1i,D}(0) = \mathbf{v}_{2i,D}(0) = 0, \mathbf{w}_{21}(0) = \mathbf{w}_{12}(0) = 0, \mathbf{g}_{1i,D}(0) = \mathbf{g}_{2i,D}(0) = 0, \mathbf{a}_{1i,D}(0) = \mathbf{a}_{2i,D}(0) = E_0, R_{1i,D}(0) = R_{3i,D}(0) = 0, R_{2i,D}(0) = R_{4i,D}(0) = E_0$
Décomposition en sous-bandes
$p_{1i}(n) = \mathbf{h}_i^T(n)\mathbf{p}_1(n), p_{2i}(n) = \mathbf{h}_i^T(n)\mathbf{p}_2(n), \text{ avec } i = 1, \dots, M,$
Prédiction
Erreurs de prédiction de sous-bande:
$\varepsilon_{1i,D}(k) = v_{2i,D}(k) - A_{1i,D}v_{2i,D}(k-1), \varepsilon_{2i,D}(k) = v_{1i,D}(k) - A_{2i,D}v_{1i,D}(k-1)$
Coefficients de prédiction de sous-bande:
$A_{1i,D}(k) = \frac{R_{1i,D}(k)}{R_{2i,D}(k)+c_b}, A_{2i,D}(k) = \frac{R_{3i,D}(k)}{R_{4i,D}(k)+c_b}$
$R_{1i,D}(k) = \eta_b R_{1i,D}(k-1) + v_{2i,D}(k)v_{2i,D}(k-1), R_{2i,D}(k) = \eta_b R_{2i,D}(k-1) + v_{2i,D}^2(k)$
$R_{3i,D}(k) = \eta_b R_{3i,D}(k-1) + v_{1i,D}(k)v_{1i,D}(k-1), R_{4i,D}(k) = \eta_b R_{4i,D}(k-1) + v_{1i,D}^2(k)$
Variances des erreurs de prédiction aller de sous-bande:
$a_{1i,D}(k) = \lambda_b a_{1i,D}(k-1) + \varepsilon_{1i,D}^2(k), a_{2i,D}(k) = \lambda_b a_{2i,D}(k-1) + \varepsilon_{2i,D}^2(k)$
Vecteurs gains de Kalman duaux de sous-bande:
$\begin{bmatrix} \tilde{\mathbf{k}}_{1i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_{1i,D}(k)}{\lambda_b a_{1i,D}(k-1)+\xi} \\ \tilde{\mathbf{k}}_{1i,D}(k-1) \end{bmatrix}, \begin{bmatrix} \tilde{\mathbf{k}}_{2i,D}(k) \\ * \end{bmatrix} = \begin{bmatrix} -\frac{\varepsilon_{2i,D}(k)}{\lambda_b a_{2i,D}(k-1)+\xi} \\ \tilde{\mathbf{k}}_{2i,D}(k-1) \end{bmatrix}$
Variables de vraisemblance de sous-bande:
$\Gamma_{1i,D}(k) = \frac{1}{1-\tilde{\mathbf{k}}_{1i,D}^T(k)v_{2i,D}(k)}, \Gamma_{2i,D}(k) = \frac{1}{1-\tilde{\mathbf{k}}_{2i,D}^T(k)v_{1i,D}(k)}$
Vecteurs gains d'adaptation de sous-bande:
$\mathbf{g}_{1i,D}(k) = \Gamma_{1i,D}(k)\tilde{\mathbf{k}}_{1i,D}(k), \mathbf{g}_{2i,D}(k) = \Gamma_{2i,D}(k)\tilde{\mathbf{k}}_{2i,D}(k)$
Estimation des sous-signaux de sortie
$v_{1i,D}(k) = p_{1i,D}(k) - \mathbf{w}_{21}^T(k-1)v_{2i,D}(k), v_{2i,D}(k) = p_{2i,D}(k) - \mathbf{w}_{12}^T(k-1)v_{1i,D}(k)$
Adaptation
Equations de mise à jour des filtres adaptatifs
$\mathbf{w}_{21}(k) = \mathbf{w}_{21}(k-1) - \mu_{21} \sum_{i=1}^M [v_{1i,D}(k)\mathbf{g}_{1i,D}(k)], \mathbf{w}_{12}(k) = \mathbf{w}_{12}(k-1) - \mu_{12} \sum_{i=1}^M [v_{2i,D}(k)\mathbf{g}_{2i,D}(k)],$
Synthèse
Signaux de sortie en plein bande
$v_1(n) = \sum_{i=1}^M \mathbf{g}_i^T(n)\mathbf{V}_{1i}(n), v_2(n) = \sum_{i=1}^M \mathbf{g}_i^T(n)\mathbf{V}_{2i}(n)$

NB: La réalisation de cet algorithme est effectuée sans décimation.

4.4. Evaluation de la complexité de calcul

La complexité de calcul de l'algorithme 3 et l'algorithme 4 est résumée dans le tableau 4. 3, dans ce tableau, la complexité de calcul inclue le nombre des multiplications et division.

Tableau 4. 3 : Complexité de calcul des deux algorithmes (Algorithme 3 et Algorithme 4).

Algorithmes	Nombre de Mul/Div
Algorithme 3	$\frac{1}{M}(4L + 22) + M(2L + 4)$
Algorithme 4	$\frac{1}{M}(4L + 22) + M(2L + 4)$

L est la longueur des filtres adaptatif et M est le nombre de sous-bandes.

4.5. Résultats de simulation des algorithmes bi-capteurs en sous-bande

Dans cette section, une validation expérimentale des deux algorithmes bicapteurs en sous-bande proposés i.e. Algorithme 3, Algorithme 4 sera présenté.

Dans nos simulations, l'environnement acoustique est modélisé par le modèle simplifié du mélange convolutif bicapteurs décrit au chapitre précédent. Ce modèle étant rappelé sur la figure 4.9, nous avons également utilisé les mêmes signaux sources de parole $s(n)$ et de bruit $b(n)$ et les mêmes réponses impulsionnelles $h_{21}(n)$ et $h_{12}(n)$ comme ceux présentés dans la section 3.6.1.

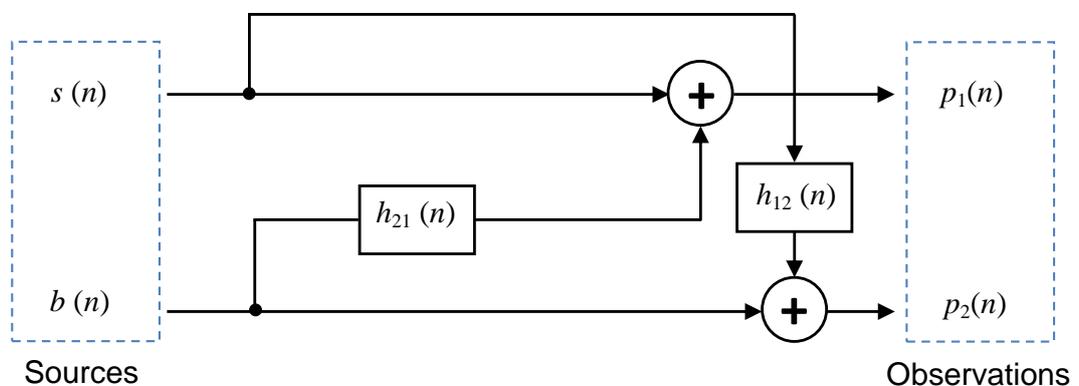


Figure 4. 9 : Modèle de mélange.

Dans les deux algorithmes bicapteurs en sous-bande proposés, nous avons utilisé d'une part un banc de filtres d'analyse pour décomposer les signaux d'observations $p_1(n)$ et $p_2(n)$ du mélange de la figure 4.9 en M sous-bandes. D'autre part, un banc de filtres de synthèse est utilisé pour reconstruire les deux ensembles des sous-signaux estimés de parole et de bruit dans leur forme pleine bande. Dans toutes les simulations présentés ci-après, nous avons choisi le nombre de sous-bandes égal à $M = 2$ et 4, tandis que la longueur des filtres de sous-bandes est choisie égale à $l = 16$ et 32 respectivement. La figure 4.10 présente les différentes sous-bandes utilisées dans les simulations des deux algorithmes i.e. Algorithme 3, Algorithme 4. Notons que nous avons utilisé les mêmes filtres de sous-bandes pour les deux bancs de filtres d'analyse et de synthèse.

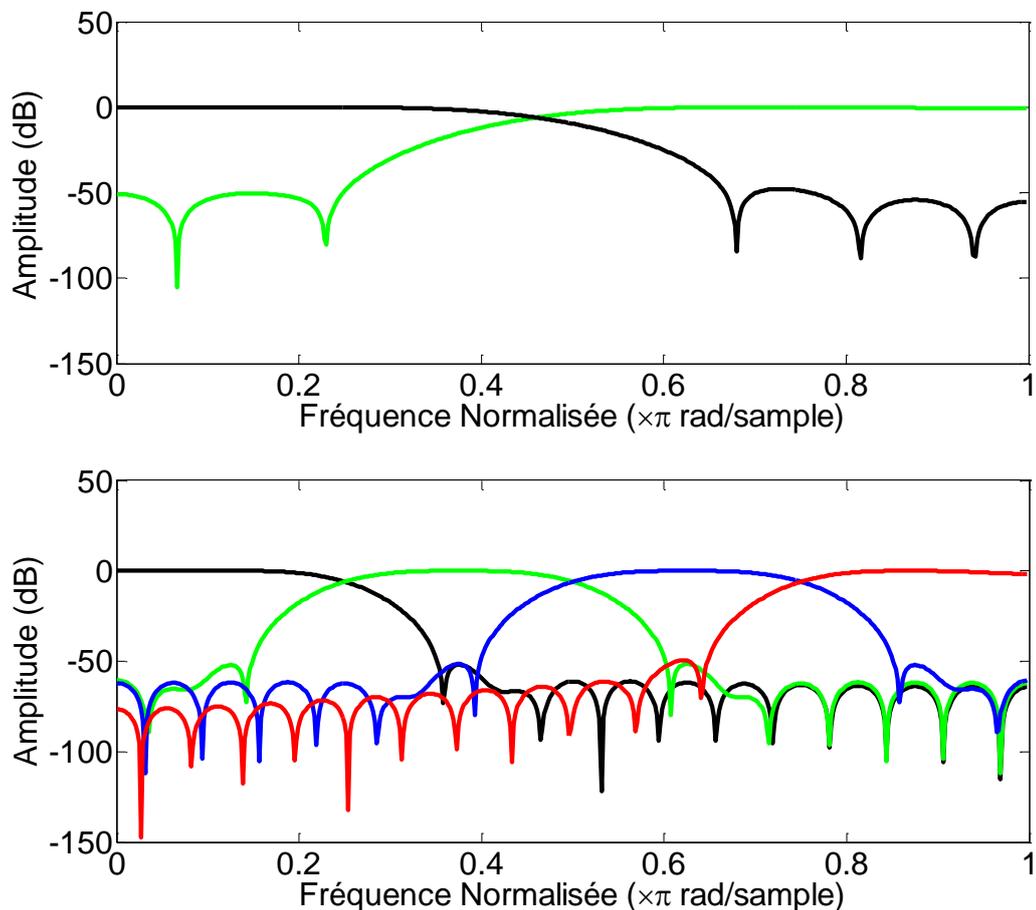


Figure 4. 10 : Caractéristique fréquentielle des filtres d'analyse et de synthèse: [en haut] pour 2 sous-bandes et [en bas] pour 4 sous-bandes.

A titre de rappel, les deux algorithmes en sous-bande proposés, Algorithme 3 et Algorithme 4 utilisent un mécanisme de détection d'activité vocale manuel pour contrôler l'adaptation des filtres $w_{21}(n)$ et $w_{12}(n)$.

4.5.1. Evaluation de l'Algorithme 3

Afin d'évaluer les performances de réduction du bruit de l'Algorithme 3, nous allons réaliser une étude comparative de ses performances avec celles de sa version pleine bande i.e. Algorithme 1.

Dans cette étude, nous nous intéresserons uniquement à la voie restituant le signal de parole utile.

Les expériences rapportées dans cette section ont été effectuées en termes de l'évaluation temporelle des signaux de sortie, le rapport signal à bruit segmental, le désajustement, l'erreur quadratique moyenne segmentale et la distance cepstrale. Ces critères ont été définis dans la section 3.6.3.

Les paramètres de simulation des deux algorithmes simulés, i.e. Algorithme 1 et Algorithme 3 sont résumés dans le tableau 4.4.

Tableau 4. 4 : Paramètres de simulation de l'Algorithme 1 et l'Algorithme 3.

Paramètres de simulation
Signaux d'entrées
Fréquence d'échantillonnage: $f_e = 8kHz$
RSB d'entrée: $RSB_1 = RSB_2 = 0,3dB$
Longueur des filtres réels: $L = 128$
Algorithme 3
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.7$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_f = 0.99$
Facteur d'oublie $\eta_f = 0.9985$
Constantes positives: $c_o = 0.001, c_f = 1$
Constante d'initialisation: $E_0 = 0.5$
Nombre de sous-bandes: $M = 2, 4$
Longueur des filtres de sous-bandes pour $M = 2$: $l = 16$
Longueur des filtres de sous-bandes pour $M = 4$: $l = 32$
Algorithme 1
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.7$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_f = 0.99$
Facteur d'oublie $\eta_f = 0.9985$
Constantes positives: $c_o = 0.001, c_f = 1$
Constante d'initialisation: $E_0 = 0.5$

- *Évaluation des signaux de sortie*

Dans cette partie, nous montrerons sur la figure 4.11 l'évolution temporelle du signal de parole originale et celle des signaux de sortie obtenus par l'Algorithme 1 (version pleine bande) et l'Algorithme 3 avec $M = 2$ et 4 sous-

bandes. Les paramètres de simulation de chaque algorithmes sont les mêmes que ceux donnés par le tableau 4.4.

En examinons cette figure, nous pouvons remarquer que les signaux de sortie obtenus par les deux algorithmes i.e. Algorithme 1 et Algorithme 3 avec 2 et 4 sous-bandes, sont presque totalement débruités, et sont très proches du signal de parole originale.

Comme première conclusion, nous pouvons dire que les deux algorithmes réduit le bruit acoustique en sortie de traitement, cependant ne nous pouvons tirer aucune conclusion sur celui qui est le bon en termes de vitesse de convergence et de distorsion. Pour cette raison, nous allons présenter dans ce qui suit, des expériences intensives en termes de d'autres critères objectifs de qualité pour bien quantifier et qualifier les performances de l'Algorithme 3 proposé par rapport à sa version pleine bande (Algorithme 1).

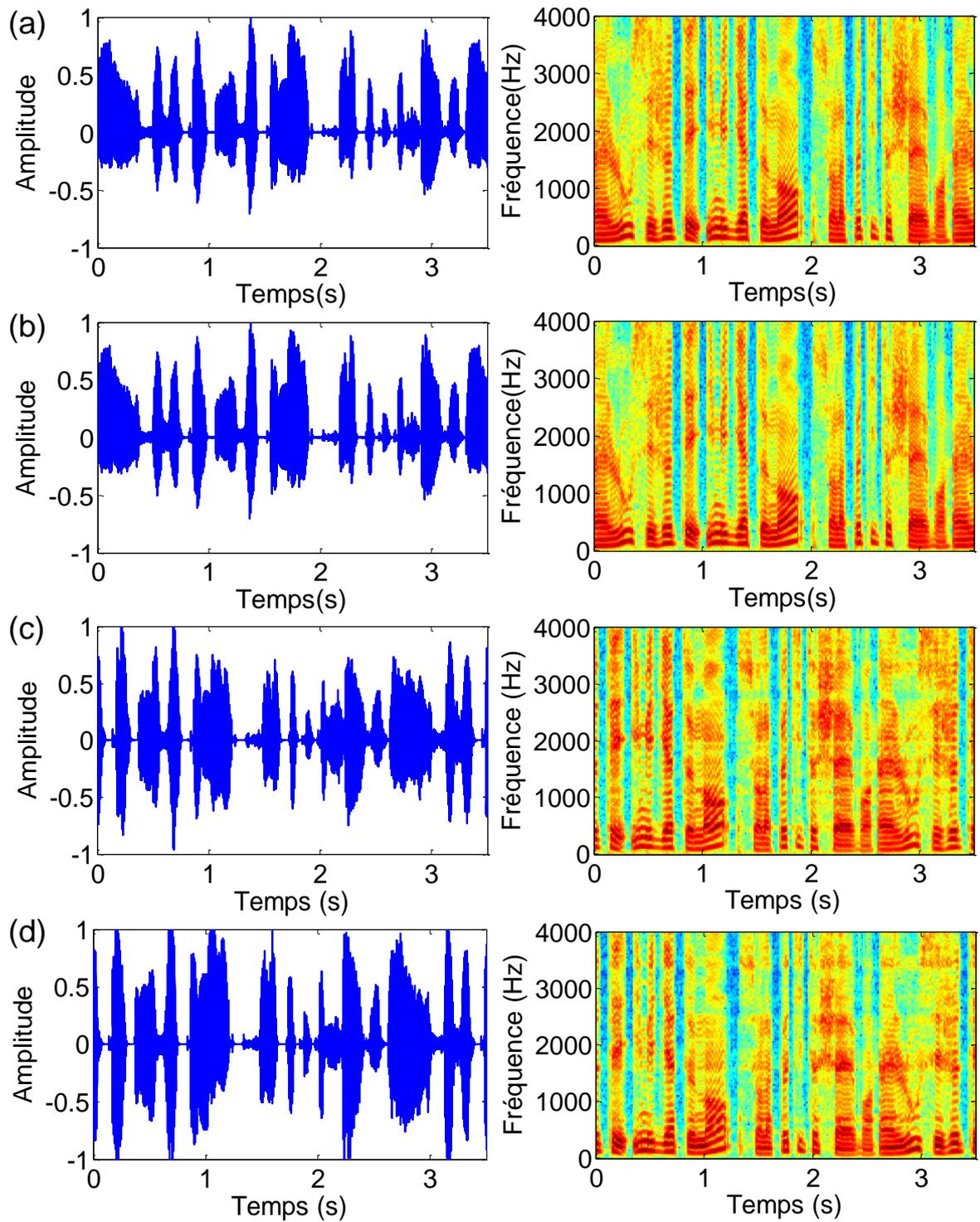


Figure 4. 11 : Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par l'Algorithme 1, (c) signal de sortie obtenu par l'Algorithme 3 avec $M = 2$, (d) signal de sortie obtenu avec l'Algorithme 3 avec $M = 4$.

- *Evaluation du désajustement*

Dans cette simulation, nous comparerons les performances de vitesse de convergence de l'algorithme 3 proposé à sa version pleine bande (Algorithme 1), en termes du critère de désajustement.

Pour réaliser cette comparaison, nous avons tracé l'évolution temporelle du désajustement (System Mismatch) pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0, 3dB$. Le signal de parole est corrompu par les composantes du bruit USASI. Ce bruit est utilisé pour évaluer l'amélioration apportée par l'algorithme 3 proposé en termes de vitesse de convergence. La figure 4.12 illustre les résultats obtenus. Notons que les paramètres de simulation sont donnés par le tableau 4.4.

Cette expérience démontre bien, que l'Algorithme 3 proposé (avec $M = 2$ et 4) comparé à sa version pleine bande, maintient une bonne performance de vitesse de convergence, en régime transitoire, et ceci pour les deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

Il est possible de voir que l'amélioration de la performance de vitesse de convergence devient plus importante lorsque le nombre de sous-bandes est choisi égal à $M = 4$

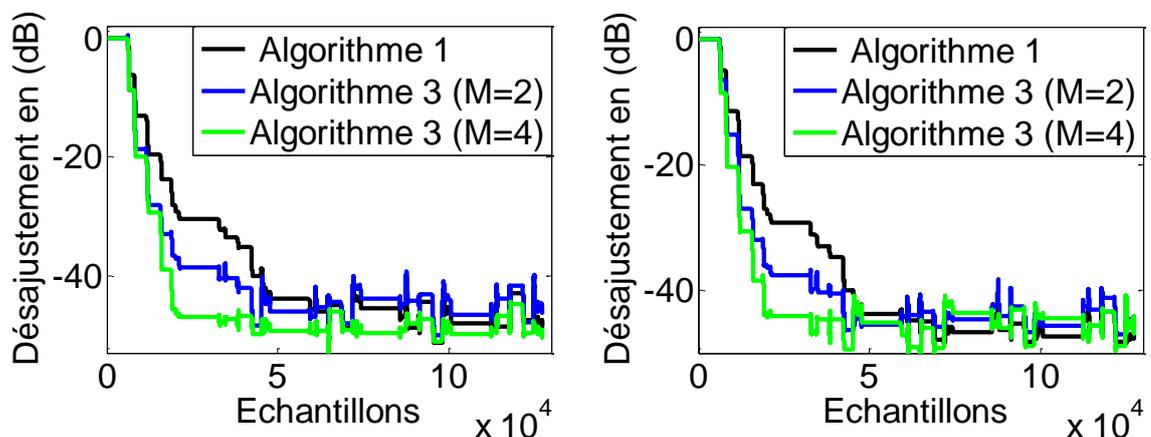


Figure 4. 12 : Evaluation du critère de désajustement de l'algorithme 3 proposé

($M = 2$ et 4) et l'algorithme 1, pour deux niveaux de RSB d'entrée:

[à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de l'erreur quadratique moyenne segmentale*

Pour confirmer les résultats précédents, nous présenterons sur la figure 3.13 l'évolution de l'EQM segmentale obtenue avec l'algorithme 3 proposé (avec $M = 2$ et 4) et sa version pleine bande Algorithme 1, pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

D'après les résultats obtenus de cette simulation, nous avons tiré les mêmes constations que celles de la section précédente, i.e. l'algorithme 3 proposé présente une vitesse de convergence nettement meilleure que celle de sa version pleine bande (Algorithme 1), surtout lorsque le nombre de sous-bande est égal à $M = 4$, et ceci pour les deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

Cette bonne propriété de vitesse de convergence s'explique par le fait que la procédure de décomposition en sous-bandes décorrèle les signaux d'entrée entre chaque sous-bande.

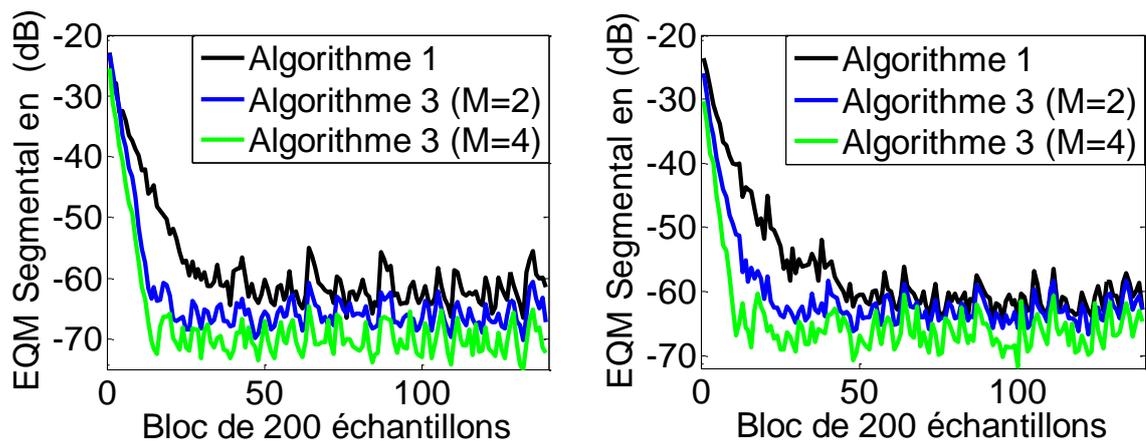


Figure 4. 13 : Evaluation du critère d'EQM segmental de l'algorithme 3 proposé (avec $M = 2$ et 4) et l'algorithme 1, pour deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation du rapport signal à bruit segmental*

La figure 3.14 compare l'évolution du critère rapport signal à bruit segmental des deux algorithmes, i.e. Algorithme 3 et Algorithme 1, pour différents niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

D'après les résultats obtenus, nous constaterons que le comportement de l'Algorithme 3 proposé est presque similaire à celui de l'Algorithme 1, avec une

légère dégradation en régime permanent. Cette dégradation augmente avec l'augmentation du nombre de sous-bandes, et ceci pour toutes les situations étudiées. Cependant, les valeurs moyennes du RSB segmental de l'Algorithme 3 proposé (avec $M = 2$ et 4) sont supérieures à $40dB$, ce qui confirme le bon comportement de cet algorithme pour les applications de réduction du bruit acoustique.

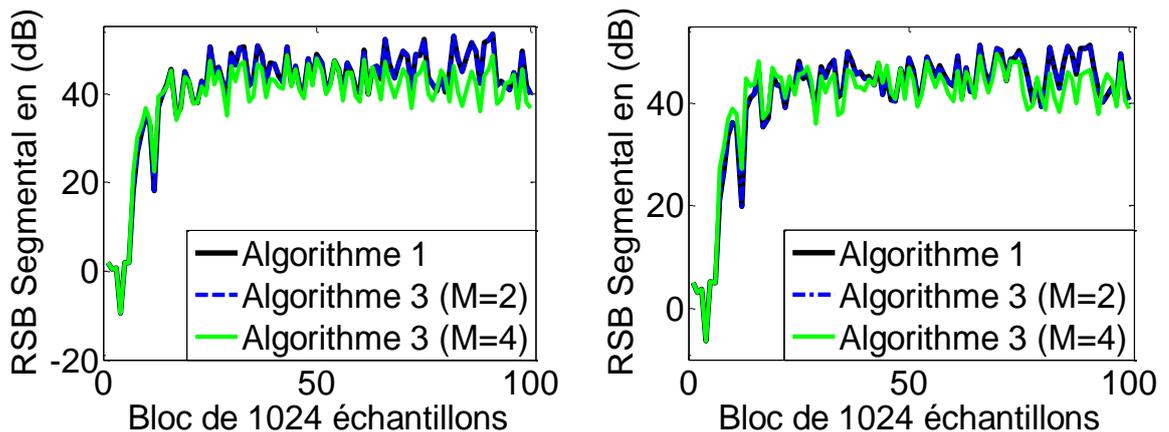


Figure 4. 14 : Evaluation du critère RSB segmental de l'algorithme 3 proposé (avec $M = 2$ et 4) et l'algorithme 1, pour deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de la distance cepstrale*

Cette dernière expérience a pour but d'analyser la distorsion apporté par l'algorithme 3 proposé en comparaison avec sa version pleine bande, i.e. Algorithme 1. Pour cela nous avons mesuré la distance cepstrale entre le signal de parole originale et celui obtenu en sortie de traitement de chaque algorithme, i.e. Algorithme 1 et Algorithme 3 (avec $M = 2$ et 4). Les résultats obtenus pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$, sont reportés sur la figure 3.15

Conformément aux résultats obtenus, nous avons remarqué que pour toutes les situations bruyantes, les valeurs finales de la distance cepstrale obtenues à partir de l'algorithme 1 (version pleine bande) sont meilleures que celles obtenues avec l'algorithme 3 proposé (avec $M = 2$ et 4). L'écart de performances entre l'algorithme 1 et l'algorithme 3 est plus significatif avec un nombre de sous-bandes élevés, i.e. $M = 4$. C'est l'inconvénient majeur de l'algorithme 3 proposé.

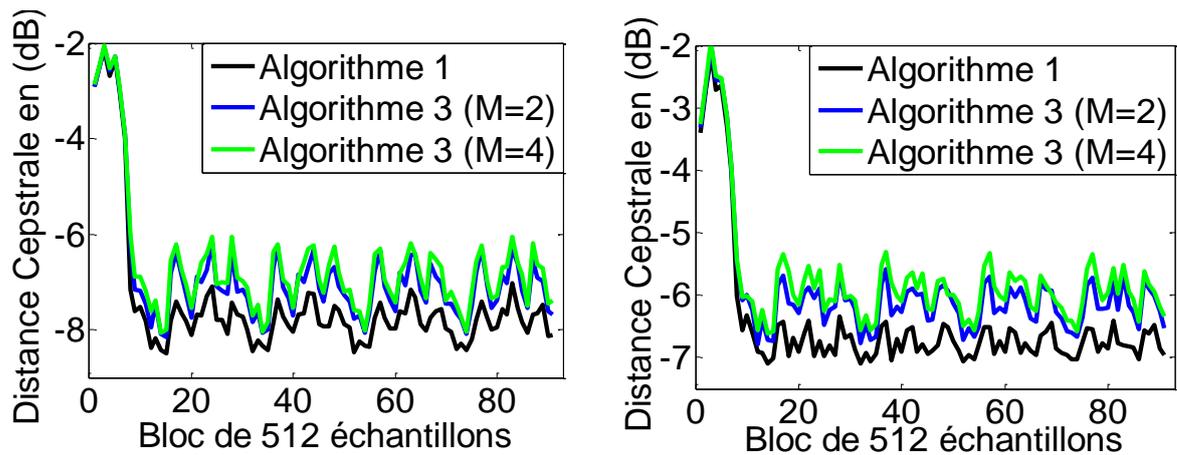


Figure 4. 15 : Evaluation du critère de la DC de l'algorithme 3 proposé (avec $M = 2$ et 4) et l'algorithme 1, pour deux niveaux de RSB d'entrée:

[à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

4.5.2. Evaluation de l'Algorithme 4

Dans cette section, Plusieurs simulations numériques dans différentes conditions seront réalisées pour évaluer et comparer les performances de l'algorithme 4 proposé. Pour ces simulations, l'algorithme proposé est comparé à sa version pleine bande (Algorithme 2), en termes de plusieurs métriques objectives. Les paramètres de simulation de chaque algorithme simulé sont listés dans le tableau 4.5.

Tableau 4. 5 : Paramètres de simulation de l'Algorithme 2 et l'Algorithme 4.

Paramètres de simulation
Signaux d'entrées
Fréquence d'échantillonnage: $f_e = 8kHz$
RSB d'entrée: $RSB_1 = RSB_2 = 0,3 dB$
Longueur des filtres réels: $L = 128$
Algorithme 4
Pas d'adaptation des filtres adaptatifs: $\mu_{12} = \mu_{21} = 0.7$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_b = 0.99$
Facteur d'oubli : $\eta_b = 0.9985$
Constantes positives: $\xi = 0.001, c_b = 1$
Constante d'initialisation: $E_0 = 0.5$
Nombre de sous-bandes: $M = 2, 4$
Longueur des filtres de sous-bandes pour $M = 2$: $l = 16$
Longueur des filtres de sous-bandes pour $M = 4$: $l = 32$
Algorithme 2
Pas d'adaptation des filtres adaptatifs: $\mu_{21} = \mu_{12} = 0.7$
Longueur des filtres adaptatifs: $L = 128$
Facteur d'oubli exponentiel: $\lambda_b = 0.99$
Facteur d'oubli : $\eta_b = 0.9985$
Constantes positives: $\xi = 0.001, c_b = 1$
Constante d'initialisation: $E_0 = 0.5$

▪ *Évaluation des signaux de sortie*

Dans cette partie, nous comparerons les signaux de sortie obtenus avec l'Algorithme 4 proposé et sa version pleine bande Algorithme 2. Les paramètres de simulation de chaque algorithme sont similaires à ceux résumés dans le tableau 4. 5. Notons que le signal de parole originale est corrompu par les composantes du bruit USASI et le niveau de RSB d'entrée est fixé à $RSB_1 = RSB_2 = 0dB$ sur les deux voie d'observations.

Sur la figure 4.16, nous montrerons l'évolution temporelle du signal de parole originale et celle des signaux de sortie obtenus par chaque algorithme, i.e. Algorithme 2 et Algorithme 4 (avec $M = 2$, et 4). Selon cette figure, nous pouvons voir que les composantes du bruit acoustique sont partiellement éliminées des signaux disponibles en sortie de traitement de chaque algorithme. L'examen par spectrogramme confirme cette dernière remarque.

Dans les tests qui suivent, nous allons évaluer les performances de chaque algorithme en termes de: le désajustement, l'erreur quadratique moyenne segmentale, rapport signal à bruit segmental, et la distance cepstrale.

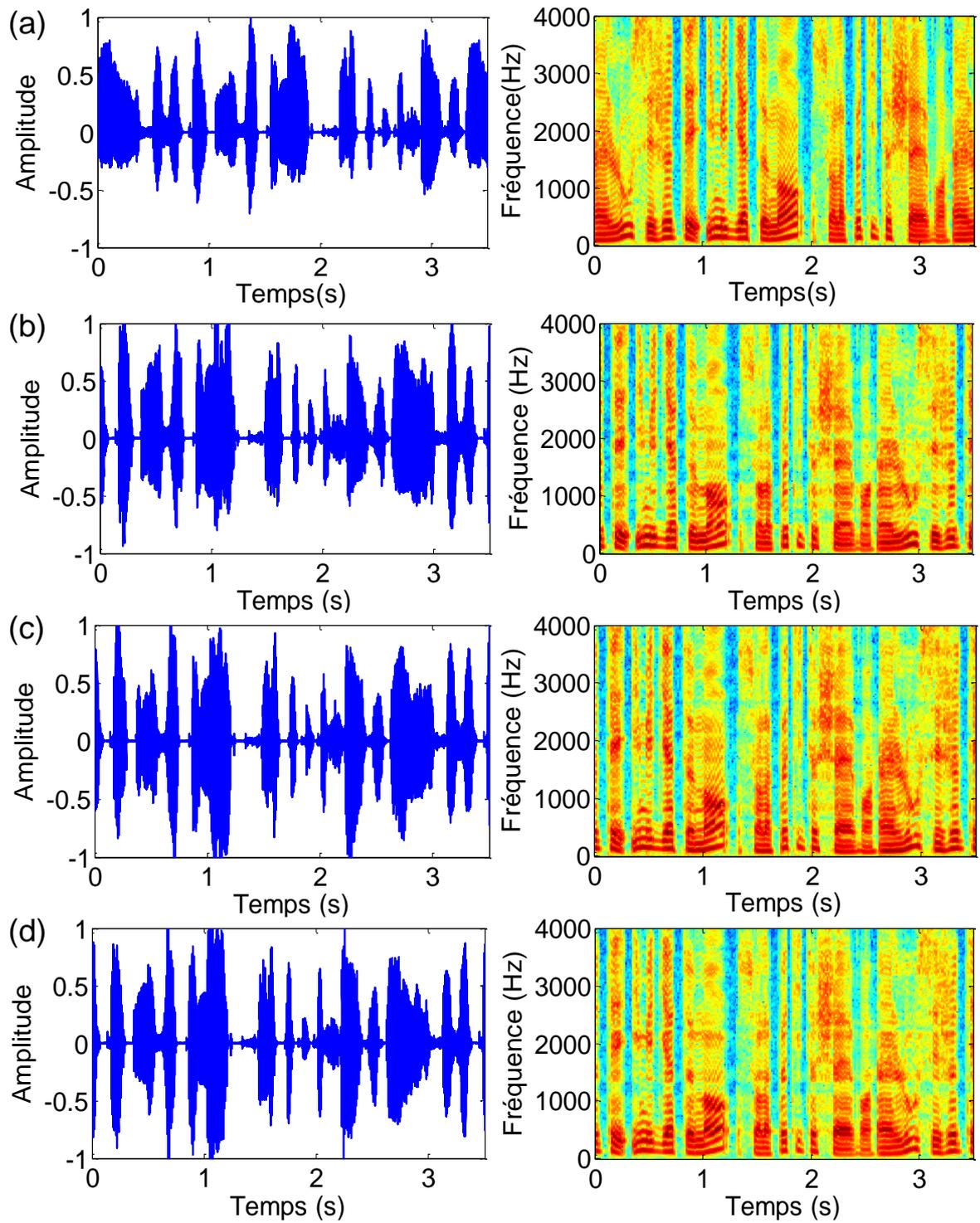


Figure 4. 16 : Evolution temporelle du: (a) signal de parole originale, (b) signal de sortie obtenu par l'Algorithme 2, (c) signal de sortie obtenu par l'Algorithme 4 avec $M = 2$, (d) signal de sortie obtenu avec l'Algorithme 4 avec $M = 4$.

- *Evaluation du désajustement*

La figure 4.17 représente une comparaison de la performance de vitesse de convergence de l'algorithme 4 proposé ($M = 2$ et 4) avec sa version pleine bande (Algorithme 2), basée sur l'évolution temporelle du désajustement mesuré sur les coefficients du filtre adaptatif $w_{21}(n)$, pour différents niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

Les résultats obtenus, montrent l'avantage de l'algorithme 4 proposé (avec $M = 2$ et 4) qui présente une vitesse de convergence rapide vers l'optimum par rapport à sa version pleine bande (Algorithme 2).

L'algorithme 4 proposé présente la plus grande vitesse de convergence lorsque le nombre de sous-bandes est élevé, i.e. $M = 4$, ce qui démontre que la vitesse de convergence de l'algorithme 4 est proportionnelle au nombre de sous-bandes.

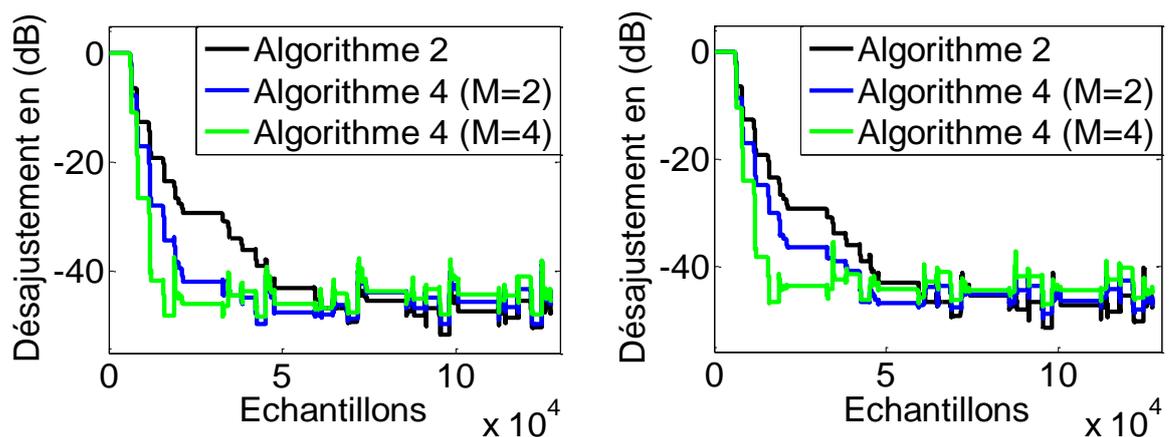


Figure 4. 17 : Evaluation du critère de désajustement de l'algorithme 4 proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux de RSB d'entrée:
[à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de l'erreur quadratique moyenne segmentale*

Pour mieux voir le comportement de vitesse de convergence de l'algorithme 4 proposé (avec $M = 2$ et 4 sous-bandes) par rapport à sa version pleine bande (Algorithme 2), nous illustrerons sur la figure 3.18, les résultats de l'évolution temporelle d'EQM segmentale de ces deux algorithmes, pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

En examinant cette figure, on remarque que l'algorithme 4 proposé (avec $M = 2$ et 4) surpasse l'algorithme 2. L'écart de performances entre l'algorithme 4 et l'algorithme 2 est plus important avec un nombre de sous-bandes élevé, i.e. $M = 4$.

Les résultats de l'évaluation du désajustement et d'EQM segmentale prouvent que l'algorithme 4 proposé permet une nette amélioration des performances de vitesse de convergence de l'Algorithme 2 (version pleine bande).

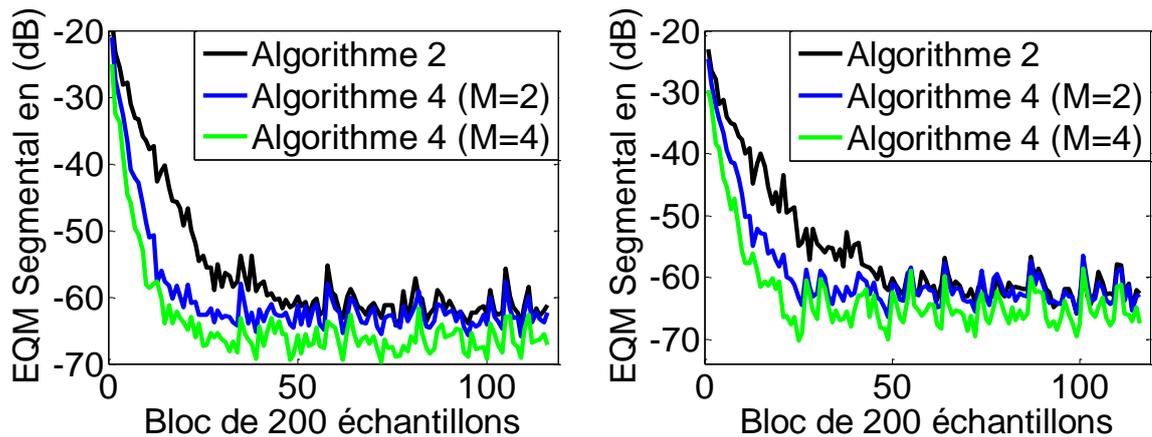


Figure 4. 18 : Evaluation du critère d'EQM segmental de l'algorithme 4 proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation du rapport signal à bruit segmental*

De la même manière que l'expérience précédente, nous évaluons l'algorithme 4 proposé (avec $M = 2$ et 4 sous-bandes) et l'algorithme 2 en termes du critère rapport signal à bruit segmental. La figure 4.19, illustre l'évolution temporelle du RSB segmental des deux algorithmes, pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

Une certaine superposition apparaît en régime transitoire des tracés du RSB segmental de l'algorithme 4 proposé (avec $M = 2$ et 4) et sa version pleine bande (Algorithme 2). De plus, une dégradation en régime permanent des valeurs du RSB segmental de l'algorithme 4 proposé est notée, lorsque le nombre de sous-bandes est égal à $M = 4$, et ceci pour les différentes situations bruyantes étudiées.

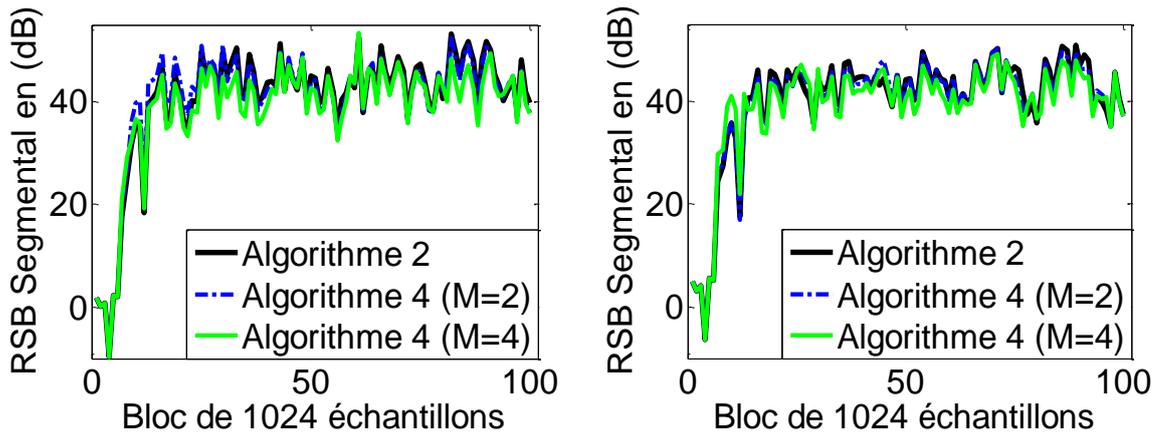


Figure 4. 19 : Evaluation du critère RSB segmental de l'algorithme 4 proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

- *Evaluation de la distance cepstrale*

Dans ce dernier test, Nous avons utilisé la mesure de la distance cepstrale pour évaluer objectivement la qualité du signal de parole rehaussé en sortie de chaque algorithme, i.e. Algorithme 4 (avec $M = 2$ et 4 sous-bandes) et Algorithme 2. La figure 4.20 présente les résultats obtenus pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

Les résultats de cette expérience montrent que les valeurs finales de la DC obtenues à partir de l'algorithme 4 proposé avec $M = 2$ sont légèrement plus faible par rapport à celles de l'algorithme 2. L'écart de performance entre l'algorithme 2 et l'algorithme 4 est plus élevé lorsque le nombre de sous-bandes est égal à $M = 4$. Cependant les valeurs moyennes de la DC de l'algorithme 4 proposé avec ($M = 2$ et 4) sont d'environ -6 dB, ce qui indique une bonne propriété d'intelligibilité du signal de parole rehaussé.

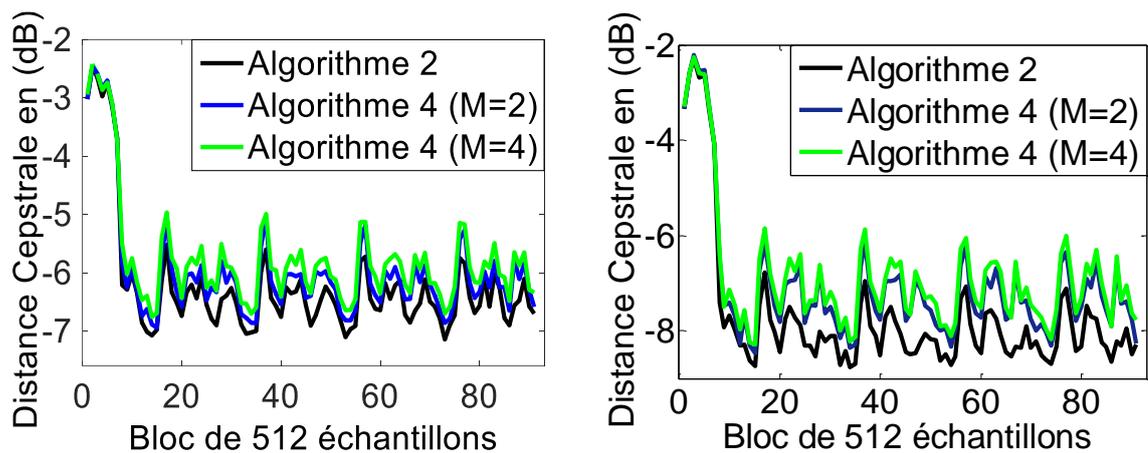


Figure 4. 20 : Evaluation du critère de la DC de l'algorithme 4 proposé (avec $M = 2$ et 4) et l'algorithme 2, pour deux niveaux de RSB d'entrée:
[à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

4.6. Conclusion

Dans ce chapitre, nous avons présenté deux algorithmes bicapteurs en sous-bande pour les applications de réduction du bruit acoustique et le rehaussement du signal de parole. Ces deux versions algorithmiques, dénommées Algorithme 3 et Algorithme 4 sont basées respectivement sur l'implémentation en sous-bande de l'algorithme 1 et l'algorithme 2 présentés au chapitre 3.

Plusieurs simulations ont été présentées pour évaluer les performances des deux algorithmes bicapteurs en sous-bande proposés. Les résultats obtenus en termes du désajustement, l'EQM segmental, le RSB segmental et la DC ont montré le bon comportement des deux algorithmes proposés ainsi que les performances supérieures de vitesse de convergence par rapport à celles de leurs versions pleine bande.

Cependant une dégradation des valeurs finales du RSB segmental et de la DC a été prouvée, notamment lorsque le nombre de sous-bandes sélectionné est élevé.

CHAPITRE 5

NOUVEL ALGORITHME BICAPTEURS A COMMUTATION PLEINE BANDE/ SOUS-BANDE.

5.1. Introduction

Dans [78] un algorithme NLMS bicapteurs en sous-bande, basé sur la FBSS a été proposé pour les applications de réduction du bruit acoustique et le rehaussement du signal de parole. Cet algorithme avait pour but l'amélioration des performances de vitesse de convergence de l'algorithme NLMS bicapteurs pleine bande [78]. Les résultats expérimentaux de cet algorithme ont prouvé sa supériorité en termes de vitesse de convergence par rapport à sa version pleine bande. Cependant, une dégradation des performances de ce dernier est avérée en régime permanent. Cette dégradation est directement proportionnelle au nombre de sous-bandes. Pour surmonter ce problème, nous allons proposer dans ce chapitre un algorithme bicapteurs à commutation pleine bande/ sous-bande. L'algorithme proposé est capable de basculer automatiquement entre les deux versions : pleine bande et sous-bande de l'algorithme NLMS bicapteurs, où le mécanisme de commutation est basé sur l'estimation de l'erreur quadratique moyenne. Cet algorithme proposé vise à combiner la bonne propriété de vitesse de convergence de l'algorithme NLMS en sous-bande avec les bonnes valeurs finales de l'algorithme NLMS pleine bande. Les résultats de simulations présentés dans ce chapitre vont montrer l'efficacité de cet algorithme proposé à commutation pleine bandes/ sous-bande dans le domaine de débruitage de la parole.

5.2. Algorithme proposé

Dans cette section, nous allons présenter un algorithme à commutation pleine bande/ sous-bande, dans lequel le mécanisme de commutation est basé sur l'estimation de l'erreur quadratique moyenne (EQM) [79]. Le schéma bloc de l'algorithme proposé est illustré sur la figure 5.1.

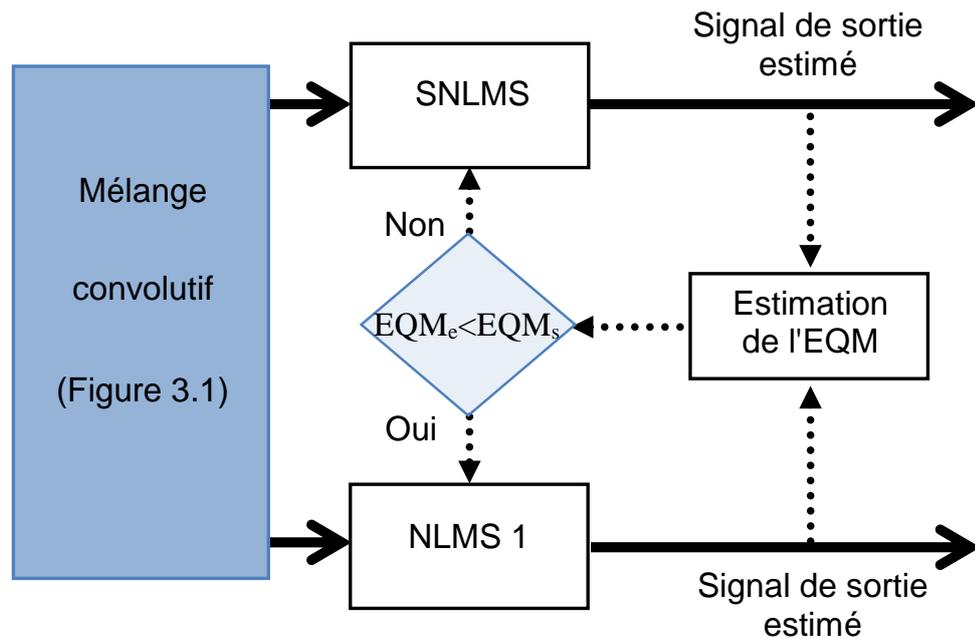


Figure 5. 1 : Schéma bloc de l'algorithme proposé (Algorithme 5).

L'algorithme proposé dénommé Algorithme 5, est capable de basculer automatiquement entre deux algorithmes : NLMS1, i.e. qui est une combinaison entre la structure de séparation aveugle de source directe et l'algorithme NLMS [52], et SNLMS, i.e. implémentation en sous-bande de l'algorithme NLMS bicapteurs [78]. D'après [78] l'algorithme SNLMS (version en sous-bande) présente une bonne propriété de vitesse de convergence par rapport à l'algorithme NLMS 1 (version en pleine bande) qui est fortement pénalisé par les non-stationnarités du signal d'entrée. Cependant l'algorithme NLMS 1 est meilleur en régime permanent (faible valeur d'EQM finale). Pour cette raison nous avons proposé de regrouper les avantages de ces deux algorithmes dans une seule structure, ce qui a donné un algorithme rapide avec des valeurs minimales en régime permanent.

L'idée principale est d'utiliser l'algorithme SNLMS pour traiter les composantes de bruit de haut niveau et d'utiliser alternativement l'algorithme NLMS1 lorsque l'intensité du bruit est faible. Le mécanisme de commutation peut être décrit par les étapes suivantes:

- Etape 1: Calcul d'énergie.

Dans cette première étape, nous calculons récursivement l'énergie $E(n)$ de l'erreur de filtrage $u_1(n)$ par la relation suivante:

$$E(n) = \beta E(n-1) + (1-\beta)u_1^2(n) \quad (5.1)$$

Où β est un facteur de lissage.

- Etape 2: Estimation de l'EQM.

Une fois l'énergie de l'erreur de filtrage est calculée, l'estimation de l'erreur quadratique moyenne peut être calculée comme suit:

$$EQM_e(n) = 10 \log_{10}(E(n)) \quad (5.2)$$

- Etape 3: Règle de commutation.

Le mécanisme de commutation entre les deux algorithmes du filtrage adaptatif, i.e. NLMS 1 et SNLMS, est effectué en comparant l'EQM estimée par rapport à un EQM seuil, La règle de commutation est définie comme suit:

$$\begin{cases} \text{Si } EQM_e < EQM_s & NLMS 1 \\ \text{Sinon,} & SNLMS \end{cases} \quad (5.3)$$

5.3. Evaluation de l'algorithme proposé

De manière à évaluer l'algorithme à commutation proposé, des simulations numériques seront réalisées dans différentes situations bruyantes. Dans l'ensemble de ces simulations, les performances de l'algorithme proposé Algorithme 5 seront comparées à celles de NLMS 1 et SNLMS. La comparaison est réalisée en termes de l'évaluation temporelle des signaux de sortie, le rapport signal à bruit segmental, le désajustement, l'erreur quadratique moyenne segmental et la distance cepstrale. Les signaux de test ont été générés de la même manière qu'au chapitre 4, nous avons également repris la même démarche de simulation et d'étude. Les paramètres de simulation de chaque algorithme sont listés dans le tableau 5.1.

Tableau 5.1: Paramètres de simulation de NLMS 1, SNLMS et Algorithme 5.

Algorithme 5	
<u>NLMS 1:</u>	<u>SNLMS:</u>
Longueur des filtres adaptatifs: $L = 128$	Longueur des filtres adaptatifs: $L = 128$
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.9$	Nombre de sous-bandes: $M = 2, 4$
<u>Mécanisme de commutation:</u>	Longueurs des filtres de sous-bandes pour $M = 2: l = 16$
L'EQM seuil: $EQMs = -45dB$	Longueurs des filtres de sous-bandes pour $M = 4: l = 32$
	Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.9$
NLMS 1	
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.9$	
Longueur des filtres adaptatifs: $L = 128$	
SNLMS	
Longueur des filtres adaptatifs: $L = 128$	
Pas d'adaptation des filtres adaptatifs: $\mu_1 = \mu_2 = 0.9$	
Nombre de sous-bandes: $M = 2, 4$	
Longueurs des filtres de sous-bandes pour $M = 2: l = 16$	
Longueurs des filtres de sous-bandes pour $M = 4: l = 32$	

5.3.1. Evaluation des signaux de sortie

Nous montrerons sur la figure 5.2, l'évolution temporelle du signal de parole originale et celle des signaux de sortie obtenus par l'algorithme NLMS 1 (version pleine bande) et l'algorithme SNLMS (version en sous-bande, avec $M = 2$) et l'Algorithme 5 proposé (algorithme à commutation pleine bande / sous-bande, avec $M = 2$). Cette expérience est réalisée avec un signal de parole $s(n)$ corrompu par un bruit USASI et de niveau de bruit fixé à $0dB$ sur les deux voies d'observations. Les paramètres de simulations de chaque algorithme sont similaires à ceux résumés dans le tableau 5.1.

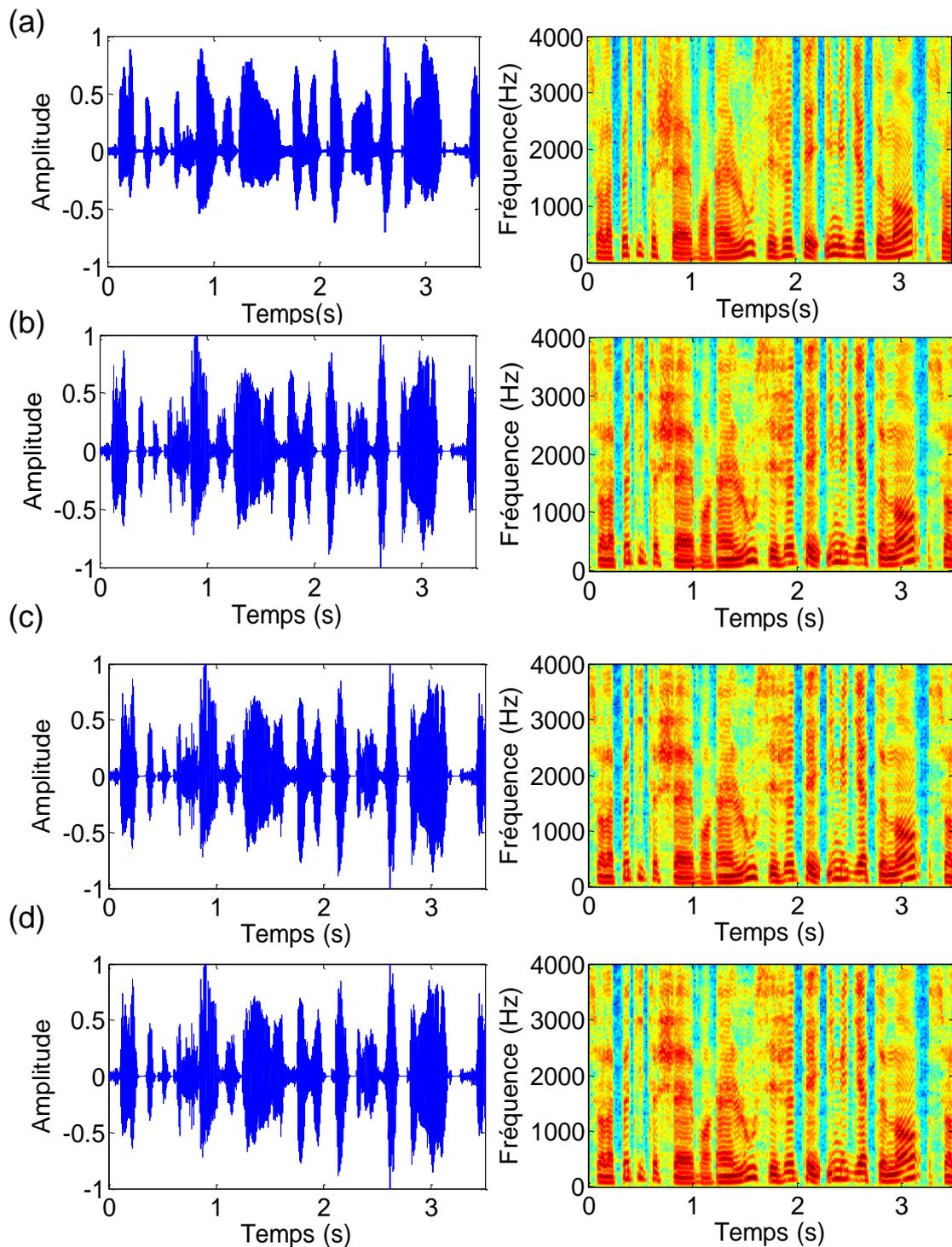


Figure 5. 2 : Evolution temporelle du: (a) signal de parole original, (b) signal de sortie obtenu par l'algorithme NLMS 1, (c) signal de sortie obtenu par l'algorithme SNLMS, avec $M = 2$, (d) signal de sortie obtenu avec l'Algorithme 5, avec $M = 2$.

À partir de cette figure, nous pouvons visuellement constater que les signaux estimés à la sortie des trois algorithmes, i.e. NLMS 1, SNLMS, et Algorithme 5,

convergent vers le signal de parole original, ce qui confirme que les composantes du bruit acoustique ont été presque totalement supprimées. Dans ce qui suit, nous allons présenter l'évaluation de quatre critères objectifs pour montrer l'amélioration apportée par l'Algorithme 5 proposé par rapport à l'algorithme NLMS 1 et l'algorithme SNLMS.

5.3.2. Evaluation du désajustement

Dans cette simulation, une comparaison entre l'algorithme 5 proposé (avec $M = 2$ et 4), l'algorithme NLMS 1 et l'algorithme SNLMS (avec $M = 2$ et 4) est faite au sens du critère de désajustement. Les résultats obtenus pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0,3dB$, sont reportés sur les figures 5.3 et 5.4.

Les résultats obtenus montrent que l'algorithme 5 proposé dépasse l'algorithme NLMS 1 et l'algorithme SNLMS (avec $M = 2$ et 4), en termes de vitesse de convergence dans les deux phases de traitement, i.e. phase transitoire et permanente. Cette bonne propriété de vitesse de convergence s'explique par le fait que l'algorithme 5 proposé, utilise l'algorithme SNLMS, lorsque l' EQM_e est supérieur à l' EQM_s , pour adapter les filtres adaptatifs, ce qui permet d'accélérer la vitesse de convergence. Cependant lorsque l' EQM_e atteint une certaine valeur seuil, l'algorithme 5 bascule vers l'algorithme NLMS 1, pour obtenir des valeurs minimales d'EQM finale.

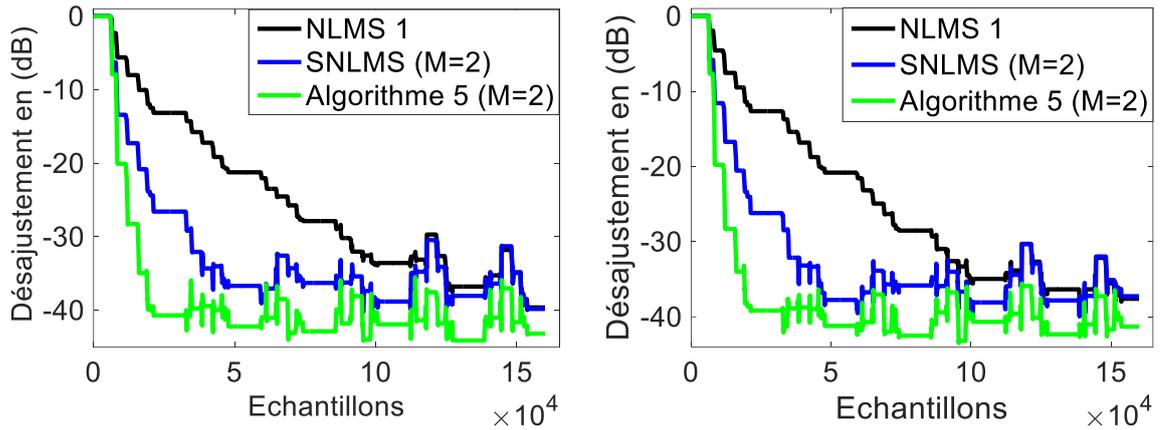


Figure 5. 3 : Evaluation du critère de désajustement de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$) , pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

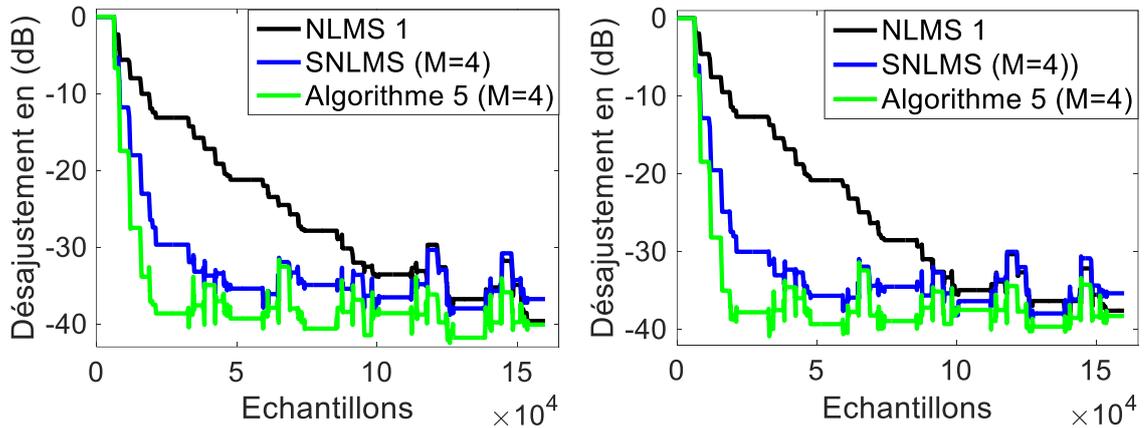


Figure 5. 4 : Evaluation du critère de désajustement de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$) , pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$ [à droite] $RSB_1 = RSB_2 = 3dB$.

5.3.3. Evaluation de l'erreur quadratique moyenne

Pour confirmer nos précédentes constatations, nous comparerons sur les figures 5.5 et 5.6, l'évolution d'EQM segmentale des trois algorithmes, i.e. NLMS 1, SNLMS (avec $M = 2$ et 4), et Algorithme 5 (avec $M = 2$ et 4), pour deux niveaux de RSB d'entrée, i.e. $RSB_1 = RSB_2 = 0$ et $3dB$. A titre de rappel, les paramètres de simulations de chaque algorithme sont similaires à ceux résumés dans le tableau 5.1.

D'après les résultats obtenus de cette simulation, nous avons tiré les mêmes constations que celles de l'expérience précédente, i.e. l'algorithme 5 proposé présente les meilleurs valeurs d'EQM segmentale en régime transitoire et permanent, ceci pour les deux niveaux de RSB d'entrée. Ces résultats prouvent bien l'amélioration du comportement de l'algorithme 5 par rapport à ceux de l'algorithme NLMS 1 (version en plein bande) et l'algorithme SNLMS (version en sous-bande). Cette amélioration est due au mécanisme de commutation pleine bande/ sous-bande.

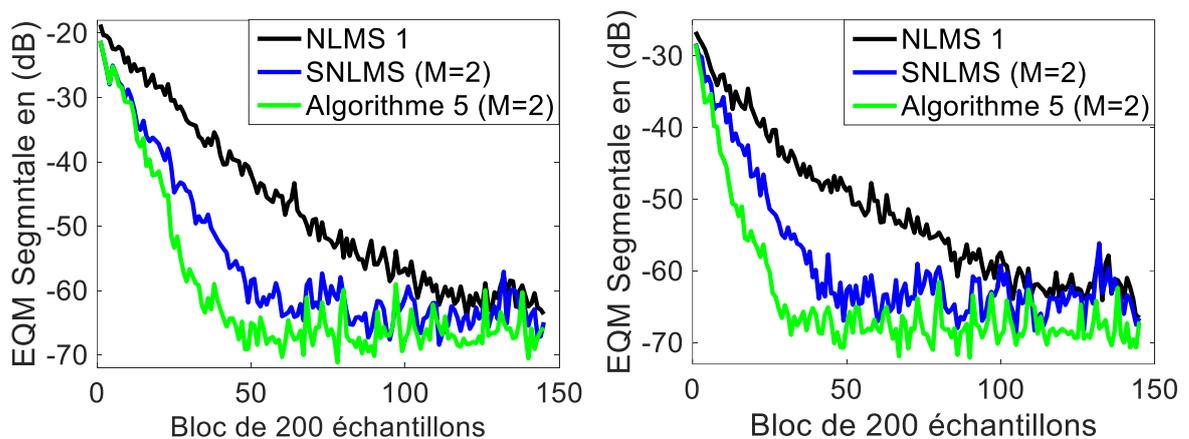


Figure 5. 5 : Evaluation du critère d'EQM Segmentale de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$) , pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

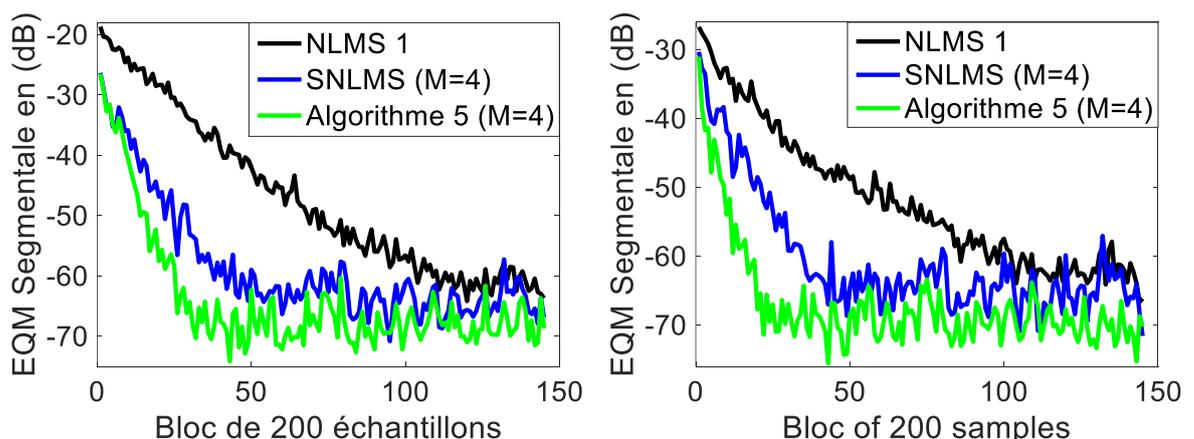


Figure 5. 6 : Evaluation du critère d'EQM Segmentale de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

5.3.4. Evaluation du rapport signal à bruit segmental

Afin d'évaluer l'amélioration apportée par l'algorithme 5 proposé sur les performances de réduction du bruit, par rapport à l'algorithme NLMS 1 et l'algorithme SNLMS, nous illustrerons sur les figures 5.7 et 5.8 l'évolution temporelle du RSB segmental des trois algorithmes pour deux niveaux de RSB d'entrée: $RSB_1 = RSB_2 = 0$ et $3dB$.

Conformément aux résultats obtenus, nous pouvons remarquer que l'algorithme proposé (avec $M = 2, 4$) se comporte plus efficacement que l'algorithme NLMS 1 et l'algorithme SNLMS (avec $M = 2$ et 4). De plus, lorsque le nombre de sous-bandes sélectionné est élevé, i.e. $M = 4$, l'algorithme proposé comparé à l'algorithme SNLMS ($M = 4$) conserve les meilleurs valeurs du RSB segmental dans les deux régimes permanent et transitoire.

Nous concluons que l'algorithme à commutation pleine bande/ sous-bande proposé a permis une nette amélioration du fonctionnement des deux algorithmes : NLMS 1 (version pleine bande) et SNLMS (version en sous-bande).

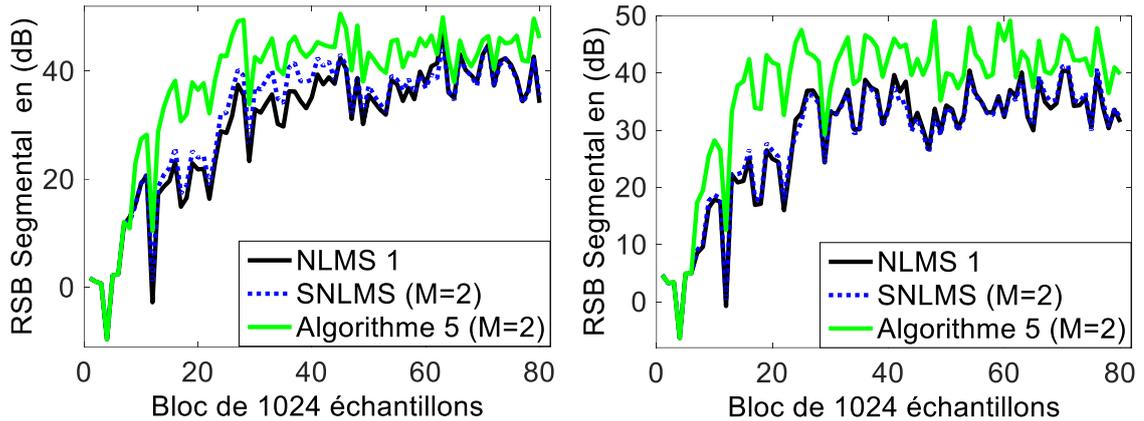


Figure 5. 7 : Evaluation du RSB Segmental de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

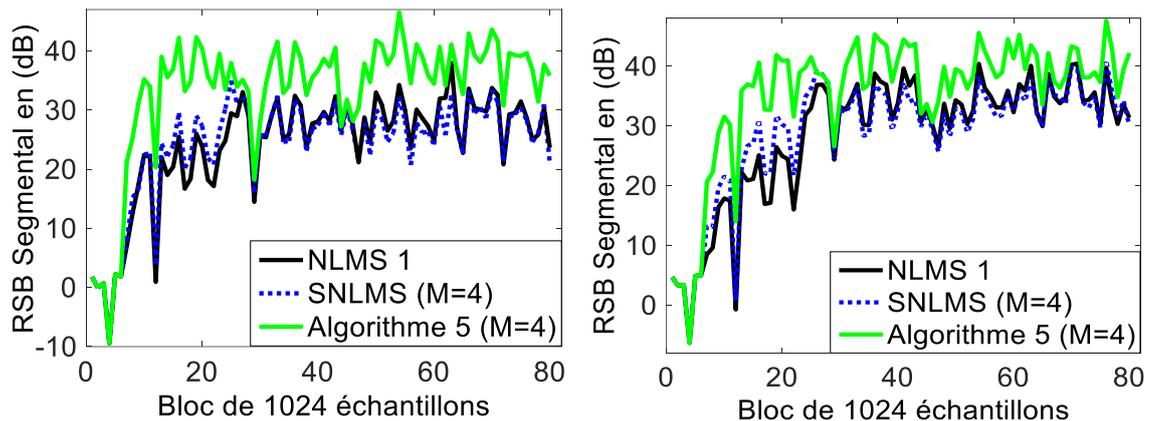


Figure 5. 8 : Evaluation du RSB Segmental de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

5.3.5. Evaluation de la distance cepstrale

Dans cette dernière simulation, nous comparerons sur Les figures 5.9 et 5.10 l'évolution temporelle de la DC des trois algorithmes, i.e. Algorithme 5 (avec $M = 2$ et 4), SNLMS (avec $M = 2$ et 4) et NLMS 1. Les conditions expérimentales sont les même que celles des sections précédentes.

D'après les résultats obtenus, nous pouvons remarquer que l'algorithme NLMS 1 dépasse l'algorithme 5 proposé et l'algorithme SNLMS, en termes de DC, et ceci pour les deux niveaux de RSB d'entrée i.e. $RSB_1 = RSB_2 = 0$ et $3dB$.

L'algorithme 5 proposé n'a pas réussi à améliorer les performances de l'algorithme SNLMS en termes de distorsion.

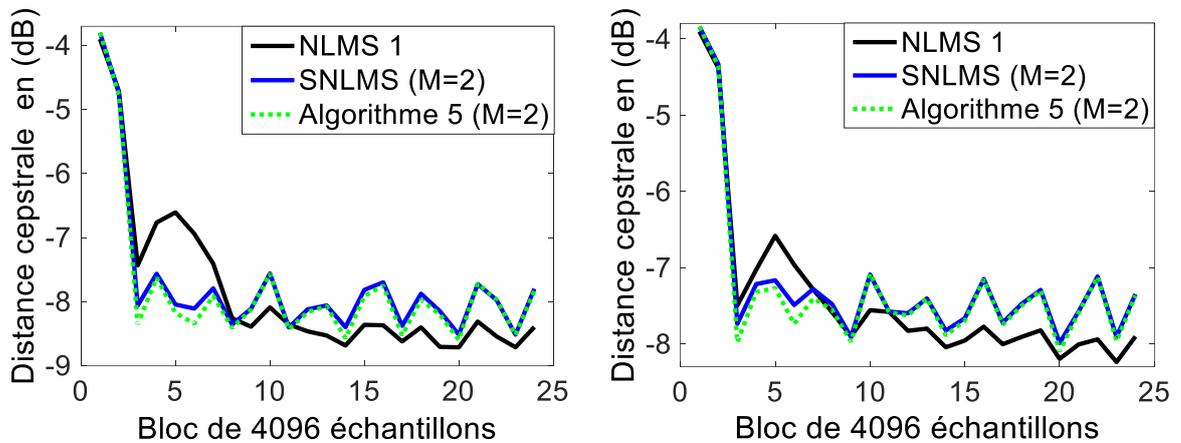


Figure 5. 9 : Evaluation de la DC de l'algorithme 5 proposé (avec $M = 2$), NLMS 1 et SNLMS (avec $M = 2$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

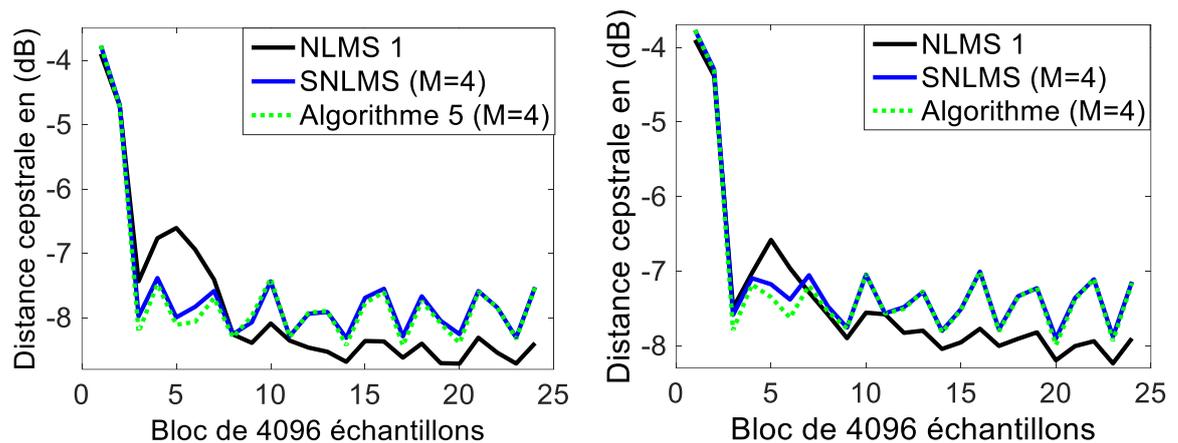


Figure 5. 10 : Evaluation de la DC de l'algorithme 5 proposé (avec $M = 4$), NLMS 1 et SNLMS (avec $M = 4$), pour un bruit USASI à l'entrée, et deux niveaux de RSB d'entrée: [à gauche] $RSB_1 = RSB_2 = 0dB$, [à droite] $RSB_1 = RSB_2 = 3dB$.

5.4. Conclusion

Dans ce chapitre, nous avons proposé un nouvel algorithme à commutation pleine bande/ sous-bande pour les applications de réduction du bruit acoustique et le rehaussement du signal de parole. Cet algorithme est capable de basculer automatiquement entre deux algorithmes: algorithme NLMS 1 (version pleine bande) et algorithme SNLMS (version en sous-bande). Le mécanisme de commutation est basé sur l'estimation de l'erreur quadratique moyenne. La

proposition de cet algorithme avait pour but l'amélioration des performances de l'algorithme SNLMS, en combinant la bonne propriété de vitesse de convergence de ce dernier avec les bonnes valeurs finales de l'algorithme NLMS 1. Les résultats de simulation ont prouvé que l'algorithme à commutation pleine bande/sous-bande proposé a permis une nette amélioration du fonctionnement de l'algorithme SNLMS en termes du RSB segmental, le désajustement et l'EQM segmentale.

CONCLUSION GENERALE

L'avènement des télécommunications mobiles a renforcé la nécessité d'améliorer la prise de son, notamment lorsqu'elle est dégradée par une perturbation qualifiée de bruit. Les travaux menés au cours de cette thèse se restreignent aux techniques bicapteurs de séparation aveugle de sources appliquées pour la réduction du bruit acoustique et le rehaussement du signal de parole.

L'objectif de notre travail était l'amélioration des performances de la réduction du bruit acoustique dans les systèmes de communication mains-libres, en particulier la téléphonie mains-libres à l'intérieur des véhicules. Dans ce contexte, nous avons proposé, comme première contribution deux algorithmes bicapteurs basés sur la séparation aveugle de sources. Le premier algorithme, repose sur la combinaison entre la structure de séparation aveugle de sources directe (FBSS) et l'algorithme du filtrage adaptatif FNLMS (fast normalized least mean square). Le deuxième algorithme est basé sur une combinaison entre la structure de séparation aveugle de sources récursive (BBSS) et l'algorithme FNLMS. La sélection de l'algorithme FNLMS est motivée par sa simplicité et ses bonnes performances en termes de vitesse de convergence. L'évaluation de ces deux algorithmes proposés a été menée par plusieurs mesures objectives. Les résultats d'une étude comparative avec d'autres algorithmes de l'état de l'art ont montrés la supériorité et le bon comportement des algorithmes proposés en termes du rapport signal à bruit segmental, l'erreur quadratique moyenne segmentale, le désajustement et la distance cepstrale.

La deuxième contribution de ce travail avait pour but l'amélioration des performances des deux algorithmes bicapteurs proposés, en termes de vitesse de convergence, par la proposition d'une implémentation en sous-bande de ces deux derniers. Les résultats de simulation ont prouvé l'amélioration apportée par les deux algorithmes en sous-bande proposés en termes de vitesse de convergence par rapport à leurs versions pleine bande. Cependant, une dégradation des

performances de ces derniers a été avérée en régime permanent, notamment lorsque le nombre de sous-bande sélectionné est élevé.

Dans la dernière partie de cette thèse, nous avons proposé un algorithme bicapteurs à commutation pleine bande/sous-bande. Cet algorithme est capable de basculer automatiquement entre les deux versions : pleine bande et sous-bande de l'algorithme NLMS bicapteurs. Le mécanisme de commutation est basé sur l'estimation de l'erreur quadratique moyenne. Les résultats obtenus avec cet algorithme sont très encourageants et montrent une nette amélioration du fonctionnement des deux algorithmes, version pleine bande et sous-bande.

Comme perspective de ce travail, nous envisagerons :

- La Proposition d'un nouvel algorithme à commutation pleine bande/ sous-bande afin d'améliorer les performances de l'algorithme 3 proposé au chapitre 3.
- L'amélioration du mécanisme de commutation par la proposition d'un seuillage automatique.
- L'implémentation de tous ces algorithmes sur des cartes DSP ou FPGA.

APPENDICE A

LISTE DES SYMBOLES ET DES ABREVIATIONS

Liste des symboles

$E[.]$: Espérance d'un signal.
$\hat{\cdot}$: Estimé d'un signal.
e^{\cdot}	: Exponentielle.
D	: Facteur de décimation.
I	: Facteur d'interpolation.
J	: Fonction coût.
f_e	: Fréquence d'échantillonnage.
$\delta(n)$: Impulsion unitaire de Kronecker.
f	: Indice de fréquence.
$Trace(\cdot)$: La somme des éléments de la diagonale d'une matrice.
$\log(\cdot)$: Logarithme népérien.
$Log_{10}(\cdot)$: Logarithme népérien à base de 10.
l	: Longueurs des filtres de sous-bandes.
C	: Nombre de capteurs.
N	: Nombre de sources.
M	: Nombre de sous-bandes.
$ \cdot $: Opérateur absolu.
$*$: Operateur de convolution.
$(\cdot)^T$: Opérateur de la transposition.
$\ \cdot\ $: Opérateur de norme euclidienne.
\sum	: Somme.
∇	: Opérateur gradient.
L	: Taille des filtres.
tr	: Temps de réverbération.

Liste des abréviations

ANC	: Annulation adaptative de bruit.
BBSS	: Backward blind source separation.
BSS	: Blind source separation.
DAV	: Détection d'activité vocale.
DC	: Distance Cepstrale.
Div	: Division.
EQM	: Erreur quadratique moyenne.
<i>EQM Seg</i>	: Erreur quadratique moyenne segmentale.
EQMe	: Erreur quadratique moyenne estimée.
EQMs	: Erreur quadratique moyenne seuil.
FBSS	: Forward blind source separation.
FNLMs	: Fast Normalized Least Mean Square.
FTF	: Fast Transversal Filter.
IFFT	: Inverse fast Fourier transform.
LMS	: Least mean square.
Mult	: Multiplication.
NLMS	: Normalized least mean square.
RIF	: Réponse impulsionnelle finie.
RLS	: Recursive least mean squares.
RSB	: Rapport signal à bruit.
RSB Seg	: Rapport signal à bruit segmental.
RSBprio	: Rapport signal à bruit <i>a priori</i> .
SAA	: Soustraction spectrale d'amplitude.
SM	: System Mismatch.
SSP	: Soustraction spectrale en puissance.
USASI	: United State of America Standard Institute.
dB	: Decibel.

REFERENCES

1. Fox, C., "Débruitage multicapteur appliqué à la téléphonie mains-libres en automobile", Thèse de Doctorat, Télécom ParisTech, France, (2013).
2. Dixit, S., Mulge, M. Y., "Review on speech enhancement techniques", *International Journal of Computer Science and Mobile Computing*, Vol 3 (8), (2014), 285-290.
3. Jin, W., Taghizadeh, M. J., Chen, K., Xiao, W., "Multi-channel noise reduction for hands-free voice communication on mobile phones", *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2017), 506–510.
4. Nabi, W., Ben Nasr, M., Aloui, N., Cherif, A., "A dual-channel noise reduction algorithm based on the coherence function and the bionic wavelet". *Applied Acoustics*, Vol 131, (2018), 186–191.
5. Upadhyay, N., & Jaiswal, R. K., "Single channel speech enhancement: Using Weiner filtering recursive noise estimation", *Procedia Computer Science*, 84, (2016), 22–30.
6. Benesty, J., Cohen, I., "Single-channel speech enhancement in the time domain", In: *Canonical correlation analysis in speech enhancement*, Springer Briefs in Electrical and Computer Engineering, (2018).
7. Nabi, W., Aloui, A., Cherif, A., "Speech enhancement in dual-microphone mobile phones using Kalman filter", *Applied Acoustics* 109 (2016), 1–4.
8. Man Kima, S., Kook Kim, H., "Noise variance estimation based on dual-channel phase difference for speech enhancement", *Digital Signal Processing*, 26, (2014), 169–182.
9. Djendi, M., Cheffi, A., "New symmetric decorrelating set-membership NLMS adaptive algorithms for blind speech intelligibility enhancement", *SN Applied Sciences*, (2019), 1:1587.

10. Al-Kindi, M.J., Dunlop, J., "Improved adaptive noise cancellation in the presence of signal leakage on the noise reference channel", *Signal processing*, vol 17, No 03, (1989), 241-250.
11. Henni, R., Djendi, M., Djebari, M., "A new efficient two-channel fast transversal adaptive filtering algorithm for blind speech enhancement and acoustic noise reduction", *Computer and Electrical Engineering*, 73, (2019), 349–368.
12. Darazirar, I., Djendi, M., "A two-sensor Gauss–Seidel fast affine projection algorithm for speech enhancement and acoustic noise reduction", *Applied Acoustics* 96, (2015), 39–52.
13. Djendi, M., "An efficient wavelet-based adaptive filtering algorithm for automatic blind speech enhancement", *International Journal of Speech Technology* 21, (2018), 355–367.
14. Djendi, M., "An efficient frequency-domain adaptive forward BSS algorithm for acoustic noise reduction and speech quality enhancement", *Computers and Electrical Engineering* 52, (2016), 12–27.
15. Philipos, C. L., "Speech enhancement, Theory and Practice", 2ème ed, CRC Press Taylor & Francis Group, (2013).
16. Haykin, S., "Adaptive Filter Theory", 5ème ed, Pearson, (2013).
17. Djendi, M., "Amélioration du comportement des algorithmes des moindres carrés transversaux rapides. Application à l'identification de réponses impulsionnelles acoustiques longues", Thèse de Doctorat Ecole Nationale de Polytechnique, Algérie, (2006).
18. Hamzé, A., "Application de la transformée en nombres entiers à la conception d'algorithmes de faible complexité pour l'annulation d'échos acoustiques". Thèse de Doctorat, Université de Bretagne Occidentale, (2007).
19. Apolinário, J.A., Netto, S.L., "Introduction to Adaptive Filters", In: QRD-RLS Adaptive Filtering. Springer, Boston, (2009).
20. Diniz, Paulo S. R., "Adaptive Filtering Algorithms and Practical Implementation". 3ème ed. Springer, (2013).
21. Michaut, F., Bellanger, M., "Filtrage adaptatif, théorie et algorithmes", Paris : Lavoisier, (Juin 2005).

22. Benesty, J., "Algorithmes du type gradient à complexité de calcul réduite et à capacité de poursuite et vitesse de convergence accrues : application à l'annulation d'écho acoustique". Rapport de recherche no193, Centre de recherches en physique de l'environnement terrestre et planétaire (CRPE). (1991), P. 190.
23. Benesty, J., Paleologu, C., Gänsler, T., Ciochină, S., "A Perspective on Stereophonic Acoustic Echo Cancellation", 1^{ère} ed. Springer-Verlag Berlin Heidelberg, Vol. 4, (2011).
24. Benveniste, A., Metivier, Priouret, P., "Algorithmes adaptatifs et approximations stochastiques : Théorie et applications à l'identification, au traitement du signal et à la reconnaissance des forme", Tome 1, Dunod, (1987).
25. Rey Vega, L., Rey, H., "A Rapid Introduction to Adaptive Filtering", 1^{ème} ed, Springer-Verlag Berlin Heidelberg, (2013).
26. Widrow, B., McCool, J. M., Larimore, M.G., Johnson, C. R., "Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter", Proceedings of the IEEE, Vol 64, n°8, (1976), 1151-1162.
27. Farhang-Boroujeny, B., "Adaptive filters: Theory and applications". 2^{ème} ed, John Wiley, UK, (2013).
28. Benallal, A., Arezki, M., "A fast convergence normalized least-mean-square type algorithm for adaptive filtering", Int. J. Control Signal Process, (2013).
29. Benesty, J., Gänsler, T., "A robust fast recursive least squares adaptive algorithm", in Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP), Salt Lake City, UT, (2001), 3785-3788.
30. Cioffi, J., Kailath, T., "Fast recursive least squares transversal filters for adaptive filtering", IEEE Trans Acoustics Speech Signal Process. ASSP-32 (April 1984), 304–337.
31. Hamidia, M., "Amélioration d'écho acoustique avec détection de la double-parole dans les communications mobiles", Thèse de doctorat, Université des sciences et de technologie Houari Bomedienne, Algérie, (2017).
32. Vaseghi, S. V., "Advanced Digital Signal Processing and Noise Reduction", 3^{ème}ed, John Wiley & Sons Ltd, Angleterre (2006).

33. Ben Aicha, A., "Réduction du bruit musical et évaluation de la qualité des signaux débruités par approches perceptuelles", Thèse de Doctorat, École supérieure des communications de Tunis, Tunisie, (2010).
34. Boll, S. F., "Suppression of acoustic noise in speech using spectral subtraction", IEEE Trans, Acoustics, Speech, Signal Processing, Vol. 27, (1979), 113-120.
35. Plapous, C., "Traitements pour la réduction de bruit. Application à la communication parlée", Thèse de Doctorat, Université de Rennes 1, France, (2005).
36. Chergui, L., "Débruitage de la parole par des méthodes basées sur les transformées discrète", Thèse de Doctorat, Université Ferhat Abbas Sétif 1, Algérie, (2017).
37. Jeanvoine, A., "Intérêt des algorithmes de réduction de bruit dans l'implant cochléaire : Application à la binauralité", Thèse de Doctorat, Université de Claude Bernard Lyon 1, Ecole Doctorale E.E.A, France, (2012).
38. Bouchibane, F. Z., "Rehaussement de la parole par la technique de formation de voies dans un environnement réel", Mémoire de Magister, Université de Blida 1, Algérie, (juin 2011).
39. Spriet, A., "Adaptive Filtering Techniques for Noise Reduction and Acoustic Feedback Cancellation in Hearing Aids", Thèse de Doctorat, Katholieke University Leuven, Belgique, (2004).
40. Chen, J., Benesty, J., Huang, Y., "A minimum distortion noise reduction algorithm with multiple microphones", IEEE transactions on audio, speech, and language processing, Vol.16, n° 3, (2008), 481-493.
41. Buchner, H., Benesty, J., Kellermann, W., "Generalized multichannel frequency-domain adaptive filtering: efficient realization and application to hands-free speech communication", Signal Process. Vol. 85, (2005), 549–570.
42. Widrow, B., Glover, J.R., Mccool, J.M., Kaunitz, J., Williams, C.S., Hean, R H., Zeidler, J.R., Dong, E.JR., Goodlin, R.C., "Adaptive Noise Cancelling: Principles and Applications", Proceedings of the IEEE, Vol. 63, n°. 12, (1975).

43. Gilloire, A., Djendi, M., Scalart, P., "Noise and distortion reduction in a forward-type structure". Brevet d'invention, Patent: WO2008037925A1, Worldwide applications, (2007).
44. Hidri, A., Meddeb, S., Abdulqadir, A., Amiri, H., "Beamforming Techniques for Multichannel audio Signal Separation", International Journal of Digital Content Technology and its Applications, vol. 6, n°. 20, (2012), 659-667.
45. Lepauloux, L., "Prise de son distante par système multimicrophone. Application à la communication parlée en environnement bruyant", Thèse de Doctorat, Université de Rennes 1, France, (2010).
46. Hérault J., Jutten C., Ans B., "Détection de grandeur primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé ", 10e colloque GRETSI sur le traitement du signal et ces application, Nice, France, (1985), 1017-1022.
47. Aissa El Bey, A., "Séparation aveugle de sources audio dans le contexte automobile", Thèse de Doctorat, Ecole nationale supérieure des télécommunications Paris, France, (2007).
48. Jarboui L., "Méthodes avancées de séparation de sources applicables aux mélanges linéaires quadratiques 2*2", Thèse de Doctorat, Université Paul Sabatier - Toulouse III, France, Université de Sfax, Tunisie, (2017).
49. Ikhlef, A., "Séparation aveugle de sources dans les systèmes de communication MIMO", Thèse de Doctorat, Université de Rennes 1, France, 2008.
50. Brahmi, A., "Contribution à la séparation de sources cyclo-stationnaires : application aux signaux de télécommunications, mécaniques et biomécaniques", Thèse de Doctorat, Université de Lyon, France, Université Sidi Mohamed ben Abdellah, Fès, Maroc, (2017).
51. Djendi, M., Bendoumia, R., "A new efficient two-channel backward algorithm for speech intelligibility enhancement: A subband approach", Applied Acoustics 76, (2014), 209–222.
52. Van Gerven, S., Van Compernelle, D., "Feed forward and Feedback in a symmetric adaptive noise canceller: stability analysis in a simplified case", Proc. IEEE EUSIPCO, Brussels, Belgium, 24-27 Aug, vol.1, (1992), 1081-1084.

53. Djendi, M., Gilloire, A., Scalart, P., "New frequency domain post-filters for noise cancellation using two closely spaced microphones", Proc. EUSIPCO, Poznan, 3-8 Sep, vol.1, (2007), 218-221.
54. Thumchirdchupong, H., Tangsangiumvisai N., "A two-microphone noise reduction scheme for hands-free telephony in a car environment", 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (2013).
55. Nabi, W., Aloui, N., Cherif, A., "An improved speech enhancement algorithm for dual-channel mobile phones using wavelet and genetic algorithm", Computers and Electrical Engineering, (2017), 1–14.
56. Djendi, M., "An efficient wavelet-based adaptive filtering algorithm for automatic blind speech enhancement". Int J Speech Technol 21, (2018), 355–367.
57. Gabrea M., "Double affine projection algorithm-based speech enhancement algorithm. In: Proceedings of the IEEE ICASSP, vol. 1, Montréal, Canada, (2003), 904–7".
58. Sayoud, A., Djendi M., Medahi, S., Guessoum, A., "A dual fast NLMS adaptive filtering algorithm for blind speech quality enhancement". Applied Acoustics 135 (2018), 101–110.
59. Sayoud, A., Djendi M., Guessoum, A., "A two-channel fast NLMS algorithm for speech enhancement and acoustic noise reduction". IEEE 5th International Conference on Electrical Engineering (ICEE), Boumerdes, Algérie, (2017).
60. Sayoud, A., Djendi M., Guessoum, A., "A Two-Sensor Fast Adaptive Algorithm for Blind Speech Enhancement", Proceeding of the 4th International Conference on Engineering & MIS (ICEMIS), Istanbul, Turquie , (2018).
61. Combescure, P., "20 listes de dix phrases phonétiquement équilibrées. Revue d'Acoustique", 56, (1981). 34–38.
62. Djendi, M., Gilloire, A., Scalart P., "Noise cancellation using two closely spaced microphones: experimental study with a specific model and two adaptive algorithms", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toulouse, France, vol.3, (2006), 744-748.

63. Quackenbush, S.R, Barnwell, T.P, Clements, M.A, "Objective Measures of Speech Quality", Prentice-Hall signal processing, 1ère edition, (1988).
64. Bosco, J., "Rehaussement de la parole combinant le domaine spectrale et le domaine des modulations du spectre", Mémoire de maîtrise, Université de Sherbrook, Québec, Canada, (2018).
65. Hu, Y., Loizou, P. C., "Evaluation of objective quality measures for speech enhancement", IEEE Transactions on Audio, Speech and Language Processing, vol 16, No 1, (2008), 229–238.
66. Ghribi, Kh., Djendi, M., Berkani, D., "A New wavelet-based forward BSS algorithm for acoustic noise reduction and speech quality enhancement". Applied Acoustics 105 (2016), 55-66.
67. Faucon, G., Le Bouquin, Akbari Azirani, R., "Mesures Objectives de la Réduction de Bruit," GRETSI, Juan-les-Pins, France, (1993).
68. Petraglia, M., Haddad, D., Marques, L., E., "Normalized Subband Adaptive Filtering Algorithm with Reduced Computational Complexity", IEEE Transactions on Circuits and Systems II: Express Briefs , Vol. 62, 12, (2015).
69. Bendoumia, R., "Annulation du bruit par les méthodes de séparation de sources aveugles. Application aux systèmes de télécommunications numériques", Thèse de Doctorat, Université de Blida 1, Blida, Algérie, (2014).
70. Choa, J., Baeka, H., Park, B., Shin, J., Variable step-size sign subband adaptive filter with subband filter selection, Signal Processing , Vol. 152, (2018), 141-147.
71. Bendoumia, R., "New sub-band proportionate forward adaptive algorithm for noise reduction in acoustical dispersive-and-sparse environments", Applied Acoustics Vol.175, (2021), 107822
72. Shi, L., Zhao, H., "A Normalized Subband Adaptive Filter with Combined Regularization Parameter", Science Direct, IFAC-PapersOnLine, 52-24 (2019), 158–162
73. Lee, K.A, Gan, W.S, Kuo, S.M., "Subband adaptive filtering: theory and implementation", 1 ère ed, Wiley 2009.

74. Djendi, M., Sayoud, A., "A new dual subband fast NLMS adaptive filtering algorithm for blind speech quality enhancement and acoustic noise reduction". *Int J Speech Technol*, Springer, 22, 391–406 (2019).
75. Araki, S., Makino, S., Aichner, R., Nishikawa, T., Saruwatari, H., "Subband based blind source separation with appropriate for each frequency band", *IEEE, ICA, Nara, Japon*, (2003), 499-504.
76. Kokkinakis, K., Loizou, P., "Subband-based blind signal processing for source separation in convolutive mixtures of speech", *Acoustics Speech and Signal Processing, IEEE*, Vol. 4, (2007), 917-920.
77. Sayoud, A., Djendi, M., "Efficient subband fast adaptive algorithm based-backward blind source separation for speech intelligibility enhancement", *Int J Speech Technol*, Springer, 23, (2020), 471–479.
78. Djendi, M., Bendoumia, R., "A new adaptive filtering subband algorithm for two-channel acoustic noise reduction and speech enhancement", *Computers and Electrical Engineering*, Elsevier, V.39, (2013), 2531–2550.
79. Sayoud, A., Djendi, M., Guessoum A., "A new speech enhancement adaptive algorithm based on fullband-subband MSE switching", *International Journal of Speech Technology*, Springer, 22, (2019), 993–1005.