

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

Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Electronique
Laboratoire Signal & Communications

Thèse de Doctorat en Electronique
Mention : Signal et Communications

Présentée par :

Mohamed DJENDI

**AMELIORATION DU COMPORTEMENT DES ALGORITHMES DES
MOINDRES CARRES TRANSVERSAUX RAPIDES. APPLICATION A
L'IDENTIFICATION DE REPONSES IMPULSIONNELLES
ACOUSTQUES LONGUES**

Devant le jurys :

Président : Mr. M. S. BOUCHERIT, Professeur à l'ENP
Examineur : Mr. H. SALHI, Professeur à l'Université de Blida.
Examineur : Mr. D. BOUKHETALA, Maître de Conférence à l'ENP,
Examineur : Mr. M. BENSEBTI, Professeur à l'Université de Blida,
Rapporteur : Mr. A. GUESSOUM, Professeur à l'Université de Blida.
Mr. D. BERKANI, Professeur à l'ENP.
Invités : Mr M. BOUCHARD, Professeur à l'Université d'Ottawa(Canada)
Mr. A. BENALLAL, Professeur à Demam (Arabie Saoudite)

Cette thèse de Doctorat est soutenue le 19 septembre 2006 à l'ENP-2006-, Alger, Algérie
« ENP : Ecole Nationale Polytechnique d'Alger »

DEDICACES

Je dédie ce modeste travail à :

Ma chère mère pour son sacrifice et ses prières ;

Mon père pour son soutien absolu ;

Mes sœurs ;

Mes frères ;

Ma grande famille ;

Tous mes amis sans exceptions ;

A aek. Maddi pour son aide et son soutien ;

A tout ceux qui cherchent le savoir partout dans le monde.

Que Allah nous aide tous à faire que le bien

Mohamed Djendi

Remerciements

Je tiens tous d'abord à exprimer ma profonde reconnaissance et mes sincères remerciements à Monsieur Abderrezak GUESSOUM, professeur à l'université de BLIDA et Directeur des nouvelles technologies au MESRS (Ministère de l'Enseignement supérieur et de la Recherche Scientifique), pour son aide déterminante et pour ses conseils précieux qui ont largement contribué dans l'avancement des travaux et l'amélioration de cette thèse.

J'adresse également mes remerciements à Monsieur Daoud BERKANI, professeur à l'ENP (Ecole Nationale Polytechnique d'Alger), d'avoir accepté la co-direction universitaire de cette thèse et d'avoir suivi et contribué avec intérêt à l'évolution de ce travail de recherche. Je le remercie également pour les corrections et les améliorations qu'il a apportées et proposées pour la rédaction de ce mémoire. Je lui exprime aussi mes remerciements pour son coté humain et ses conseils

J'exprime ma gratitude à Monsieur Mohamed Seghir BOUCHERIT, Professeur à l'ENP, d'avoir bien voulu présider mon jury.

Je suis très reconnaissant à Monsieur Djamel BOUKHETALA, Maître de conférence à l'ENP, et à Monsieur Hassan SALHI, Maître de conférence à l'université Saad Dahlab de BLIDA, d'avoir accepté d'être membre de la commission d'examen.

J'exprime toute ma reconnaissance à Monsieur Messaoud BENSEBTI, Professeur à l'Université Saad Dahlab de BLIDA, qui m'a fait l'honneur d'être mon examinateur et de témoigner ainsi de l'intérêt qu'il porte à ce travail. Je tiens à le remercier pour son soutien aux moments les plus difficiles que j'ai passés à l'Université de Saad Dahlab BLIDA et pour la confiance qu'il m'a toujours accordée.

J'adresse également mes remerciements à Monsieur Martin BOUCHARD, professeur à l'université d'OTTAWA, pour sa contribution à l'évaluation de mes travaux et qui m'a permis, au travers de nombreuses discussions fructueuses d'enrichir bien des connaissances liés à l'acoustique, le filtrage adaptatif et les algorithmes des Moindres carrés rapides. Monsieur Martin Bouchard trouve ici mon témoignage pour l'aide qu'il m'a apportée dans la publication de mes travaux à ICASSP 2004 et dans le Journal de traitement du signal (Signal Processing).

Je tiens à exprimer ma gratitude et mon amitié à Monsieur Ahmed BENALLAL, professeur à DEMMAM en ARABIE SAOUDITE, pour m'avoir initié aux domaines du filtrage adaptatif, de l'Acoustique et des algorithmes des Moindres carrés rapides, pour m'avoir encadré et suivi dans tous mes projets de recherche à l'Université Saad Dahlab de BLIDA, et de m'avoir soutenu dans les moments difficiles. Qu'il trouve ici ma reconnaissance et mon respect.

Abstract/Résumé

Abstract

In this thesis, five new versions of the Fast Newton Type adaptive Filtering (FNTF) algorithm are presented. A brief preliminary presentation of these algorithms was given in ICASSP Conference 2004 and a detailed one is published in Signal Processing Journal. The first algorithm is based on a simple modification of the filtering part, by introducing a scalar accelerator parameter. The second algorithm is based on the use of the temporal subdivision technique to update the local filter coefficients. The third algorithm is a modification of the second one, by the use of the final filtering errors to update the filter coefficients. The fourth and the fifth algorithms are based, respectively, on the combination of features from the first algorithm with features of the second and third algorithms. These new algorithms are proposed to improve the convergence speed of the original version of the FNTF algorithm for the identification of acoustic impulse responses, and also to improve the tracking ability when the systems vary in time. A comparative study of each algorithm with the original version of the FNTF algorithm is also presented. In addition to these five versions two other versions of the FTF are proposed and another version per block of the simplified Newton algorithm is proposed.

Résumé

Dans cette thèse, cinq nouvelles versions de l'algorithme de filtrage adaptatif de type newton rapide (FNTF) sont présentées. Une brève présentation préliminaire de ces algorithmes a été donnée dans la conférence d'ICASSP 2004 et une autre version plus détaillée est publiée dans le journal Signal Processing. Le premier algorithme est basé sur une simple modification de la partie de filtrage par l'introduction d'un paramètre accélérateur. Le deuxième algorithme est basé sur l'utilisation d'une technique de découpage temporelle dans l'actualisation des coefficients du filtre. Le troisième algorithme est une modification du deuxième algorithme, par l'utilisation de l'erreur de filtrage finale pour actualiser les coefficients du filtre. Les quatrième et cinquième algorithmes sont basés, respectivement, sur la combinaison des propriétés du premier algorithme avec les propriétés du deuxième et le troisième algorithme. Ces nouveaux algorithmes sont proposés pour l'amélioration de la vitesse de convergence de la version originale de l'Algorithme FNTF pour l'identification des réponses impulsionnelles acoustiques, et pour améliorer aussi la capacité de poursuite lorsque le system varie dans le temps. Une étude comparative de chaque algorithme avec la version originale de l'algorithme FNTF est présentée.

ملخص
 في هذه الأطروحة، سنعرض خمسة صيغيات جديدة لالغوريتم الترشيحات المتوائمة من نوع نيوتن السريعة.
 لقد اعطينا عرض جد سطحي لكل هذه الغوريتمات في اللقاء العلمي ايكاسب 2004، و صيغة جد متطورة في الجريدة العلمية - معالجة الإشارة. الالغوريتم الأول مبني على تبديل بسيط لمرحلة الترشيح و ذلك بإدخال عامل مسرع. الالغوريتم الثاني مبني على استعمال تقنية التقسيم الزمني في تجديد عوامل المرشح. الالغوريتم الثالث هو تغيير الالغوريتم الثاني بإستعمال الخطأ في الترشيح النهائي في تجديد جميع عوامل المرشح. اما الالغوريتمان الرابع و الخامس، فهما مبنيان على التجسيم بين خصائص الالغوريتم الأول و الثاني لتجسيم الرابع، و على تجسيم خصائص الالغوريتم الثاني مع الثالث لتجسيم الالغوريتم الخامس. كل هذه الالغوريتمات المقترحة تسمح بتحسين سرعة الإنتهاء للالغوريتم الأصلي FN F، وذلك لتعريف الأجوبة النبطية الصوتية، وكذلك لتحسين قدرة المتابعة للأنظمة المتغيرة في الزمن. نقدم ايضاً دراسة بالمقارنة لكل الغوريتم جديد مع الالغوريتم الأصلي FN F. بالإضافة لهذه الخمس الالغوريتمات المقترحة و الجديدة لنيوتن، نعرض صيغتان جديدتان للالغوريتم FN F، و صيغة جدارية جديدة اخرى لالغوريتم نيوتن المبسط.

Mots clés: filtres adaptatifs, Algorithme du filtre transverse de Newton rapide (FNTF), vitesse de convergence, capacité de poursuite, systèmes acoustiques, Algorithme de filtre transverse rapide (FTF), Algorithme des Moindres carrés normalisés (NLMS).

Sommaire

Sommaire

Introduction générale	1
-----------------------------	---

Chapitre I : Généralités sur les Algorithmes de Filtrage Adaptatifs et l'Annulation d'Echo Acoustique

I.1. Introduction	5
I.2. Position du problème.....	6
I.3. Critères de comparaison	10
I.4. Présentation des algorithmes.....	10
I.4.1. Algorithmes du gradient stochastique LMS et NLMS.....	10
I.4.2. Algorithmes des moindres carrés exacts	11
I.4.2.1. Algorithme des moindres carrés récursifs (RLS)	11
I.4.2.2. Algorithmes des moindres carrés transversaux rapides (FTF)	14
I.5. Algorithme FTF 7L	20
I.6. Stabilisation numérique de l'algorithme FTF7N.....	21
I.7. Initialisation de l'algorithme FTF 7N.....	21
I.8. Les algorithmes par bloc	22
I.8.1. Les nouveaux algorithmes par blocs	22
I.8.2. Séparation entre la taille du filtre et la taille du bloc.....	22
I.8.3. Application à des algorithmes équivalents au LMS	24
I.9. Conclusion	25

Chapitre II : Présentation des Algorithmes des Moindres Carrés Transversaux Rapides de Type Newton

II.1. Introduction.....	27
II.2 Principe de base de l'algorithme FNTF.....	27
II.3. Principe de base de l'extrapolation de la corrélation.....	28
II.3.1. Modélisation statistique de la corrélation (cas stationnaire)	28
II.3.2. Modélisation récursive (Cas non stationnaire)	30
II.4. Dérivation de la version originale de l'algorithme adaptatif FNTF12N2L	34
II.5. Stabilisation numérique de l'algorithme FNTF.....	39
II.6. Initialisation de l'algorithme	40
FNTF.....	41
II.7 L'algorithme Fast Newton Simplifié (FNS)	44
II.8 Complexité des calculs.....	45
II.9 Conclusion.....	

Chapitre III : Nouvelles Versions des Algorithmes des Moindres Carrés Transversaux Rapides de Type Newton

III.1. Introduction	47
III.2. Les nouvelles versions de l'algorithme FNTF.....	48
III.2.1. Algorithme 1	48
III.2.2. Algorithme 2	49
III.2.3. Algorithme 3	51
III.2.4. Algorithme 4.....	52
III.2.5. Algorithme 5	53
III.3. Stabilisation numériques des nouveaux algorithmes	53
III.4. Évaluation de la complexité de calcul	54
III.5. Initialisation des algorithmes proposés.....	55
III.6. Analyse de la vitesse de convergence et la capacité de poursuite des cinq versions algorithmiques proposées.....	
III.6.1. Modèles de performance	56
III.6.2. Analyse de la vitesse de convergence.....	56
III.6.2.1. Algorithme 1.....	57
III.6.2.2. Algorithmes 3 et 5.....	57
III.6.3. Analyse de la capacité de poursuite.....	58
III.6.3.1. Algorithme 1	59
III.6.3.2. Algorithmes 2, 3, 4 et 5.....	61
III.7. Résultats expérimentaux des nouvelles versions de l'algorithme FNTF.....	61
III.7.1. Test de la vitesse de convergence	61
III.7.2. Test de la capacité de poursuite avec un canal variable dans le temps.....	64
III.7.3. Test de la capacité de poursuite avec un canal variable dans le temps artificiellement.....	68
III.7.4. Comparaison des performances avec un canal acoustique variable dans le temps.....	69
III.8. Algorithme FNS par bloc et simulations.....	74
III.9. Nouvelles méthodes combinatoires de l'algorithme FTF.....	76
III.10. Conclusion	77

Chapitre IV : Proposition d'Une Nouvelle Méthode Pour l'Amélioration de l'Annulation d'Echo Acoustique par les Algorithmes Adaptatifs

IV.1 Introduction.....	79
IV.2 Principe de base de l'annulation d'écho acoustique.....	80
IV.3 Description des critères de performances.....	81
IV.4 Acquisition de la réponse impulsionnelle.....	82
IV.5 Résultats de simulations.....	83
IV.5.1 Algorithme NLMS Modifié (NLMSmr)	83
IV.5.2 performance du nouvel algorithme modifié NLMSmr.....	83
IV.5.3 Résultat de simulation des algorithmes NLMS et NLMSmf	89
IV.5.4 Etude des paramètres de l'algorithme FNTF VERSION 1	93
IV.5.5 Modification de l'algorithme FNTF VERSION 1 (FNTF VERSION1mr)...	94
IV.5.6 Résultats de simulation de l'algorithme FNTF VERSION 1 et FNTFVERSION 1mr.....	94

IV.6 Conclusion	98
Conclusion générale	101
Bibliographie	103
Biographie	

Liste d'abréviations

LMS: Least Mean Square
MCR: Moindres Carrées Rapides
BLMS: Bloc Least Mean Square
NLMS: Normalized Least Mean Square
FTF: Fast Transversal Filter
FNTF: Fast Newton Transversal Filter
FNS: Fast Newton Simplified
FIR: Finite Impulse Response
IIR: Infinite Impulse Response
EQM: Erreur Quadratique Moyenne
MSE: Mean Square Error
RLS: Recursive Least Square
MA: Moyenne Ajustée
ARMA: Autorégressive à Moyenne ajustée
FTF7L: Fast transversal Filter (7L est la complexité de calcul et L est la taille du filtre)
MDF : Multidelay bloc frequency Domain adaptive Filter
FELMS: Fast Exact Least Mean Square
UMDF : Upper Multidelay bloc frequency Domain adaptive Filter
FD-LMS : Fourier domain Least Mean Square
AR : Autoregressive
FNTF12N2L : Fast Newton Transversal Filter ($12N+2L$: représente la complexité de calcul ou N est la taille des prediteurs, et L est la taille du filtre)
LS : Least square
FRLS : Fast Recursive Least square
URLS : Upper Recursive Least Square
BFTF8L : Bloc Fast transversal Filter avec une complexité de 8L (L est la taille du filtre)
BFNTF12N2L : Bloc Fast Newton Transversal Filter avec une complexité de $12N+2L$ (N :taille des prediteurs et L est la taille du filtre)
BFNS : Bloc Fast Newton Simplified
BNLMS : Bloc Normalised Least Mean Square
NLMSmr : Normalised Least Mean Square Modifié par récurrence
NLMSmf : Normalised Least Mean Square Modifié par Fenêtre
RSB : Rapport signal sur bruit
SNR : Signal ton noise Ratio

Introduction générale

Le téléphone et les réseaux informatiques sont les moyens de communication essentiels dans les télécommunications à longue distance. Dans le système téléphonique classique, le message parlé est capté par le combiné puis transmis au locuteur lointain. Le combiné téléphonique et le canal acoustique sont les seules responsables de la qualité de la communication. Le phénomène d'écho électrique qui peut apparaître dans ce système est la conséquence de la réflexion d'une partie du signal émis en un point où l'adaptation n'est pas parfaite. Ce phénomène d'écho devient très gênant pour les liaisons téléphoniques à grande distance (exemple : communication par satellite) [1].

Dans les systèmes de télécommunication, on considère la salle ou l'habitacle de voiture comme un maillon à part entière de la chaîne de transmission. Ils doivent faire face aux problèmes liés à la prise de son qui réduisent l'intelligibilité de la parole. Deux phénomènes sont responsables de cette dégradation [2]: la réverbération due aux multiples réflexions captées par le microphone et l'écho provenant du couplage acoustique du haut-parleur avec le microphone, souvent proche d'un terminal. La communication devient même impossible lorsque la chaîne de transmission devient instable (effet Larsen). Sans atteindre ce cas extrême, l'écho acoustique est aussi responsable d'une certaine gêne pour le locuteur lointain. En effet, celui-ci entend sa propre voix retardée par le délai de transmission. Le locuteur sera perturbé si ce délai dépasse environ 30ms.

Ces problèmes doivent être traités par les terminaux audio de prise et restitution du son. Un dispositif de traitement de signal est donc intégré dans le terminal pour atténuer ce signal d'écho. Les premières techniques dites à communication ou variation de gain ne permettant pas une communication bidirectionnelle confortable, l'utilisation d'un annuleur d'écho, éventuellement couplé avec une variation de gain plus souple, est une solution conceptuelle simple pour résoudre le problème.

L'annulation d'écho est basée classiquement sur l'estimation du canal acoustique entre le haut-parleur et le microphone par des techniques d'identification adaptatives des systèmes linéaires. Linéaire, car le canal acoustique est en première approximation composé essentiellement de retards et d'atténuations. La modélisation habituelle du canal acoustique est donc sa réponse impulsionnelle. Identification adaptative, car le canal acoustique varie dans le temps suivant les mouvements dans la salle ou dans l'habitacle de la voiture.

L'identification des réponses impulsionnelles acoustiques dans la classe générale de l'identification des systèmes linéaires est caractérisée par:

- Le nombre de coefficients à identifier est important, voire très important. Dans un contexte de téléconférence, plusieurs milliers de coefficients peuvent être nécessaires pour avoir une bonne estimation d'écho. C'est une conséquence du temps de réverbération, intrinsèquement lié au volume de la salle, qui peut dépasser 300ms dans des salles acoustiquement non traitées. En revanche, dans le contexte radio mobile, seul 200 à 300 coefficients sont nécessaires, vu le faible volume de l'habitacle de la voiture. Ces deux exemples montrent la grande variabilité de l'ordre du modèle du canal acoustique.
- Le signal d'excitation du filtre est la parole, signal corrélé et fortement non stationnaire, qui n'a pas de bonnes propriétés pour l'identification. En effet, la corrélation du signal réduit la vitesse d'apprentissage du système dans des algorithmes simples, classiquement utilisés, et les non stationnarités de ce signal perturbent aussi l'identification pendant les périodes de faible excitation en rendant l'algorithme plus sensible au bruit ambiant.
- Les non-stationnarités du canal acoustique engendrées par les mouvements dans la salle ne sont pas du type rupture, ni dérive permanente, mais plutôt une succession de périodes d'évolutions entrelacées avec des périodes stationnaires.
- Le bruit ambiant qui perturbe l'identification, peut faire évoluer le RSB (écho/bruit) de plus de 30dB en téléconférence à moins de 10dB dans une voiture.

Pour toutes ces raisons, il n'existe pas a priori d'algorithme universel pour traiter une si grande variété de situations, d'autant qu'aux contextes d'identification s'ajoutent les contraintes particulières d'intégration de l'algorithme dans le système global, comme la complexité numérique et le retard de traitement.

L'étude que nous avons menée s'est orientée vers la recherche puis la proposition d'un ou plusieurs algorithmes efficaces dans le contexte bruité de l'annulation d'écho dans les radios mobiles. Nous avons privilégié la vitesse de convergence et la capacité de poursuite sur la complexité de calcul ; nous restreignons notre recherche aux algorithmes purement récursifs effectuant une mise à jour à chaque nouvelle information.

Le premier chapitre de cette thèse présente un état de l'art non exhaustif de l'identification adaptative directe appliquée à l'annulation d'écho. Nous rappelons les deux algorithmes récursifs les plus connus, le LMS (Least Mean Squares) et les MCR (Moindre Carrés Récursifs). Nous présentons ensuite une nouvelle génération d'algorithmes traitant des algorithmes par blocs classique (BLMS: Block LMS) tout en accélérant la vitesse d'acquisition du filtre. On décrit aussi les algorithmes FTF (Fast Transversal Filter) équivalents aux RLS (Recursive Least Square) ainsi que la méthode de stabilité utilisée.

Le second chapitre décrit une nouvelle gamme d'algorithmes (FNFTF: Fast Newton Transversal Filter) récemment découverte, dont la complexité peut être fortement réduite par rapport aux FRLS (Fast Recursive Least Square). Cette simplification provient de la réduction de la partie prédiction d'ordre L à un ordre N suffisant pour une bonne décorrélation du signal d'excitation. Cette famille d'algorithmes ne repose plus sur une estimation directe de la matrice d'autocorrélation d'ordre L , mais sur celle d'ordre N suivie d'une extrapolation jusqu'à l'ordre L . Tout comme le NLMS (Normalized LMS) assimile tous les signaux comme des bruits blancs, les FNFTF les considèrent comme des processus autorégressifs d'ordre N . Une description de l'algorithme FNFTF simplifié (FNS : Fast Newton simplified) est présentée à la fin de ce chapitre.

Dans le troisième chapitre, nous présentons les nouvelles versions algorithmiques que nous proposons dans cette thèse et qui font la partie originale de cette thèse. Cinq nouveaux algorithmes de type FNTF, deux nouveaux algorithmes de type FTF, ainsi qu'une nouvelle version par bloc de l'algorithme FNS seront présentés dans ce chapitre. Une étude théorique de toutes ces versions algorithmiques ainsi que leurs résultats des simulations seront détaillés et présentés telles que la vitesse de convergence, la capacité de poursuite et la stabilité numérique.

Dans le quatrième chapitre, nous proposons une nouvelle méthode (technique) avec les algorithmes de filtrage adaptatifs, qui permet une nette amélioration de l'annulation d'écho acoustique dans les systèmes multimédia. Cette technique repose sur l'intermittence du signal de parole. Les résultats des simulations d'une version de l'algorithme de Newton FNTF (version 1) seront présentés dans ce chapitre ainsi que celle du gradient stochastique.

En conclusion, le travail réalisé dans cette thèse a permis la proposition de cinq nouvelles versions de l'algorithme FNTF classique, deux autres versions similaires à l'algorithme FTF et une nouvelle version par bloc de l'algorithme FNTF simplifié (FNS). Une technique simple et efficace est aussi proposée dans cette thèse pour améliorer l'annulation d'écho acoustique.

Chapitre I

Généralités sur les algorithmes de filtrages adaptatifs et l'annulation d'écho acoustique

I.1. Introduction

Les nouveaux services de télécommunication considèrent la salle ou l'habitacle de voiture comme un maillon à part entière de la chaîne de transmission. Ils doivent faire face aux problèmes liés à la prise de son qui réduisent l'intelligibilité de la parole. Deux phénomènes sont responsables de cette dégradation [1]: la réverbération due aux multiples réflexions captées par le microphone et l'écho provenant du couplage acoustique du haut-parleur avec le microphone, souvent proche sur un terminal. La communication devient même impossible lorsque la chaîne de transmission devient instable (effet Larsen). Sans atteindre ce cas extrême, l'écho acoustique est aussi responsable d'une certaine gêne pour le locuteur lointain.

En effet, les personnes présentes dans une salle réentendent leurs propres voix ; c'est le phénomène d'écho acoustique dû au canal acoustique de couplage. Le locuteur sera perturbé si ce délai dépasse environ 30ms (voir figure I.1)

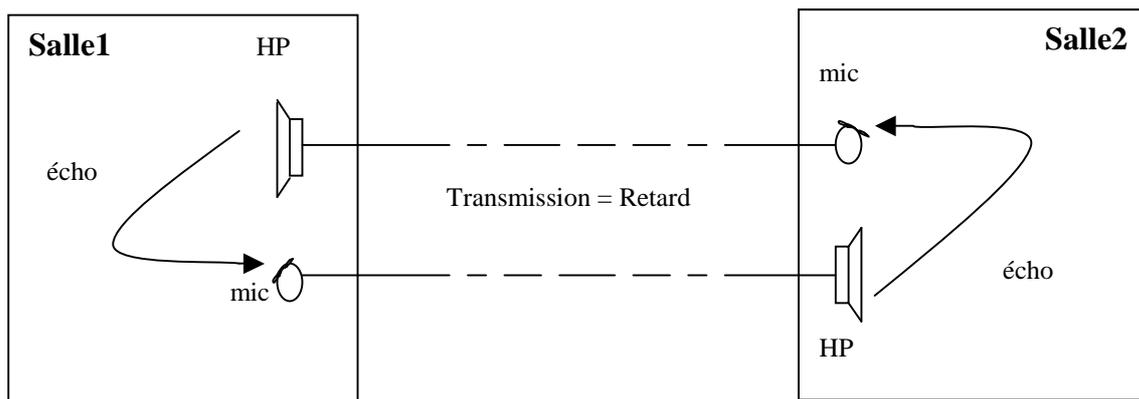


Figure I.1 : Communication bidirectionnelle entre deux salles

Par définition [2], un canal acoustique de couplage représente la transformation du signal diffusé par le haut-parleur de sonorisation et capté par les microphones de prise de son. Par ailleurs, l'approximation de l'acoustique linéaire permet d'affirmer qu'un canal acoustique

réalise un filtrage linéaire qui peut être étudié indépendamment des signaux. Il dépend essentiellement de la salle, des positions et des caractéristiques des sources et des récepteurs.

L'annulation d'écho acoustique est un problème qui se pose dans un grand nombre d'application (téléphonie main libre, téléconférence,...). Les difficultés essentielles rencontrées sont d'une part liées à la durée des réponses impulsionnelles des canaux acoustiques à identifier et d'autre part à la nature des signaux à traiter.

Les terminaux audio doivent traiter ces problèmes de prise et restitution du son. Un dispositif de traitement de signal est donc intégré dans le terminal pour atténuer ce signal d'écho. Les premières techniques dites à communication ou variation de gain ne permettant pas une communication bidirectionnelle confortable, l'utilisation d'un annuleur d'écho, éventuellement couplé avec une variation de gain plus souple, est une solution conceptuelle simple pour résoudre le problème.

Dans ce qui suit, nous allons définir le problème et présenter les algorithmes d'identification adaptative de base ainsi que la notion "complexité de calcul".

I.2. Position du problème

Le problème qui nous intéresse dans cette étude est l'estimation continue d'un signal s_t , éventuellement perturbé par un autre signal p_t , à partir d'un autre signal x_t , appelé signal de référence. La composante p_t est non mesurable, elle est incluse dans l'observation captée sur une voie dite principale : $y_t = s_t + p_t$. La composante p_t peut représenter le signal utile pour certaines applications. Par exemple : dans le cas du débruitage de la parole avec voie de référence, p_t est la parole, s_t est le bruit perturbateur et x_t le bruit seul. Dans l'annulation d'écho acoustique, en absence ou en présence de la double parole, x_t est le signal reçu du locuteur lointain, y_t est le signal d'écho du locuteur lointain vers lui même et p_t le bruit ambiant du milieu acoustique ou le signal de la double parole du locuteur proche.

Dans les deux cas, la restauration du signal utile est réalisée en soustrayant à l'observation y_t une estimée du signal s_t obtenue par un filtrage approprié du signal de référence x_t [7]. Le schéma de principe de ce filtrage est représenté sur la figure I.1, où les signaux sont connus aux instants d'échantillonnage t . La période d'échantillonnage est T_e , et l'on a : $x_t = x(t) = x(tT_e)$.

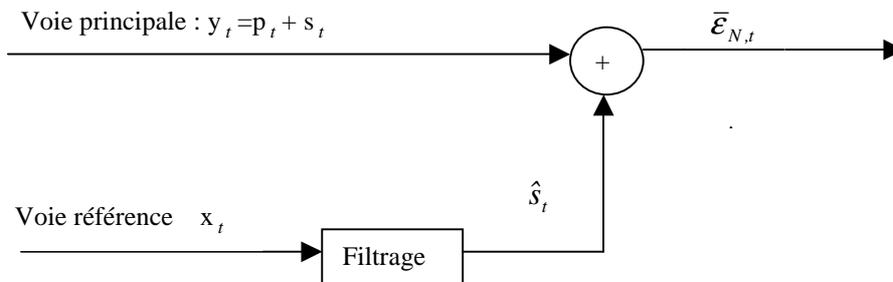


Figure I.2 : Schéma de principe d'un système d'annulation d'écho acoustique ou de débruitage de la parole.

Il faut noter que l'optimum, en annulation d'écho acoustique, est d'avoir la sortie $\bar{\varepsilon}_{L,t} = 0$, en absence de double parole ou du bruit perturbateur ($p_t = 0$), alors que cet optimum se transforme en $\bar{\varepsilon}_{L,t} = s_t$ dans le cas du débruitage de la parole.

Les hypothèses de base concernant le système de la figure I.1 sont d'une part l'indépendance entre la composante p_t et les signaux s_t et x_t , et d'autre part la corrélation des signaux s_t et x_t .

D'après le schéma de la figure I.2, l'estimation du signal s_t à partir du signal x_t revient à trouver ou à estimer le filtre H_L . Par conséquent, des applications telles que l'annulation d'écho acoustique ou le débruitage de la parole avec voie de référence bruit seul consistent essentiellement à identifier le filtre H_L . Ce dernier doit en principe modéliser le trajet physique (canal acoustique de couplage en annulation d'écho) suivi par l'onde acoustique en se propageant de la voie dite de référence vers la voie dite principale. Le filtre modélisateur H_L est soit un filtre à réponse impulsionnelle finie (FIR), soit un filtre à réponse impulsionnelle infinie (IIR).

On se place d'emblée dans le cas de l'identification d'une réponse impulsionnelle acoustique où l'identification du filtre H_L par une réponse impulsionnelle finie a déjà fait ses preuves [3]. Par conséquent, on suppose que le signal s_t se déduit de x_t par un filtrage linéaire causal. Dans ces conditions, la sortie du système de la figure I.2 s'écrit :

$$\bar{\varepsilon}_{L,t} = y_t - H_L^T X_{L,t} \quad (\text{I.1})$$

avec

$$H_L^T = [H_L^1, H_L^2, \dots, H_L^L] \quad (\text{I.1i})$$

$$X_{L,t}^T = [x_t, x_{t-1}, \dots, x_{t-L+1}] \quad (\text{I.1ii})$$

où H_N désigne un vecteur d'ordre L contenant les coefficients de la réponse impulsionnelle finie et $X_{L,t}$ désigne un vecteur qui résume le passé du signal x_t sur une durée de L échantillons. L'exposant T désigne l'opérateur de transposition.

La solution optimale du problème de l'estimation de s_t à partir d'une combinaison linéaire x_t est fournie par la solution de l'équation de WIENER-HOPF, qui exprime la nullité du gradient de l'erreur quadratique moyenne en sortie

$$\nabla_H E \left\{ \bar{\varepsilon}_{L,t}^2 \right\} \Big|_{H=H_{opt}} = 0 \quad (\text{I.2})$$

où $E\{\}$ et ∇_H désignent respectivement la moyenne statistique et l'opérateur gradient.

La solution est donnée par la relation suivante :

$$H_{L,opt} = R_{L,xx}^{-1} P_{L,xy} \quad (\text{I.3})$$

avec

$$P_{L,xy} = E\{X_{L,t}y_t\} \quad (\text{I.4})$$

$$R_{L,xx} = E\{X_{L,t}X_{L,t}^T\} \quad (\text{I.5})$$

Où le vecteur $P_{L,xy}$ désigne l'intercorrélation entre l'observation y_t et le vecteur signal $X_{L,t}$ dans la mémoire du filtre ; et la matrice carré $R_{L,xx}$, d'ordre L , désigne la matrice d'autocorrection du signal de référence dans le filtre, elle est symétrique et définie positive. Cette dernière propriété assure l'existence et l'unicité de la solution optimale.

Cependant, dans la plupart des applications pratiques, les statistiques des signaux sont inconnues a-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 le filtre H_L change en fonction des informations reçues où l'erreur en sortie est utilisée pour la mise à jour des coefficients du filtre H_L . Les coefficients du filtre sont ajustés à chaque instant t par un algorithme d'adaptation itératif. Une procédure itérative bien connue qui annule le gradient de l'erreur quadratique moyenne est donnée dans [5].

D'après [5], l'algorithme du gradient déterministe s'écrit :

$$H_{L,t} = H_{L,t-1} - \mu \left[\nabla_H E\{\bar{\epsilon}_{L,t}^2\} \right]_{H=H_{L,t-1}}$$

avec

$$\nabla_H E\{\bar{\epsilon}_{L,t}^2\} \Big|_{H=H_{L,t-1}} = 2R_{L,xx}H_{L,t-1} - 2P_{L,xy}$$

où $P_{L,xy}$ et $R_{L,xx}$ sont respectivement données par les relations (I.4) et (I.5).

En utilisant l'égalité suivante :

$$R_{L,xx}H_{L,t-1} - P_{L,xy} = E\{X_{L,t}X_{L,t}^T H_{L,t-1} - X_{L,t}y_t\} = -E\{X_{L,t}\bar{\epsilon}_{L,t}\}$$

L'algorithme du gradient déterministe s'écrit :

$$H_{L,t} = H_{L,t-1} + 2\mu E\{\bar{\epsilon}_{L,t}X_{L,t}\} \quad (\text{I.6})$$

où μ est un pas d'adaptation, éventuellement variable, qui contrôle la stabilité et la vitesse de convergence de l'algorithme et $H_{L,t}$ est un vecteur dépendant du temps t donnée par :

$$H_{L,t}^T = \left[H_{L,t}^1, H_{L,t}^2, \dots, H_{L,t}^L \right] \quad (\text{I.6i})$$

et

$$\bar{\epsilon}_{L,t} = y_t - H_{L,t-1}^T X_{L,t} \quad (\text{I.6ii})$$

où $\bar{\varepsilon}_{L,t}$ désigne l'erreur de filtrage a-priori calculée avant la mise à jour du filtre.

Le principal rôle d'un algorithme adaptatif est d'ajuster un vecteur de paramètre $H_{L,t}$ pour un objectif bien défini (minimisation de l'EQM). Le schéma de principe d'un filtre adaptatif bouclé par un algorithme d'adaptation est représenté sur la figure I.3.

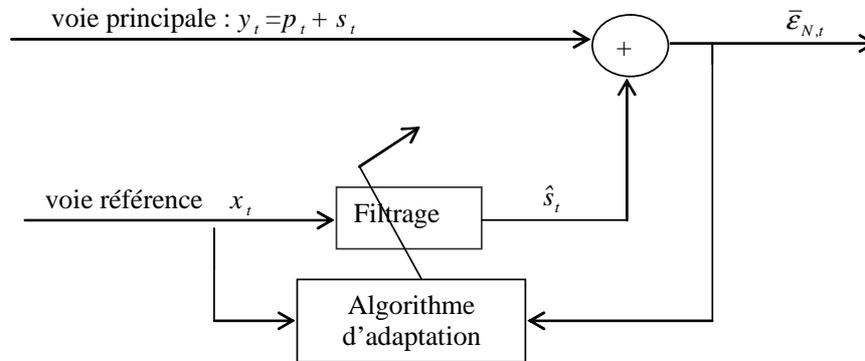


Figure I.3 : Schéma de principe d'un système d'annulation d'écho acoustique

Pour étudier le schéma de la figure I.3, diverses approches peuvent être utilisées.

Elles correspondent d'une part à la structure du filtre adaptatif et d'autre part à la technique utilisée pour adapter les coefficients du filtre adaptatif. Les algorithmes adaptatifs ont été largement étudiés dans la littérature, il existe plusieurs ouvrages traitant des toutes sortes d'algorithmes adaptatifs [4],[6],[7],[14].

Par conséquent, la structure sera bien évidemment transversale. Le modèle de filtre transverse est donné par le schéma de la figure I.4.

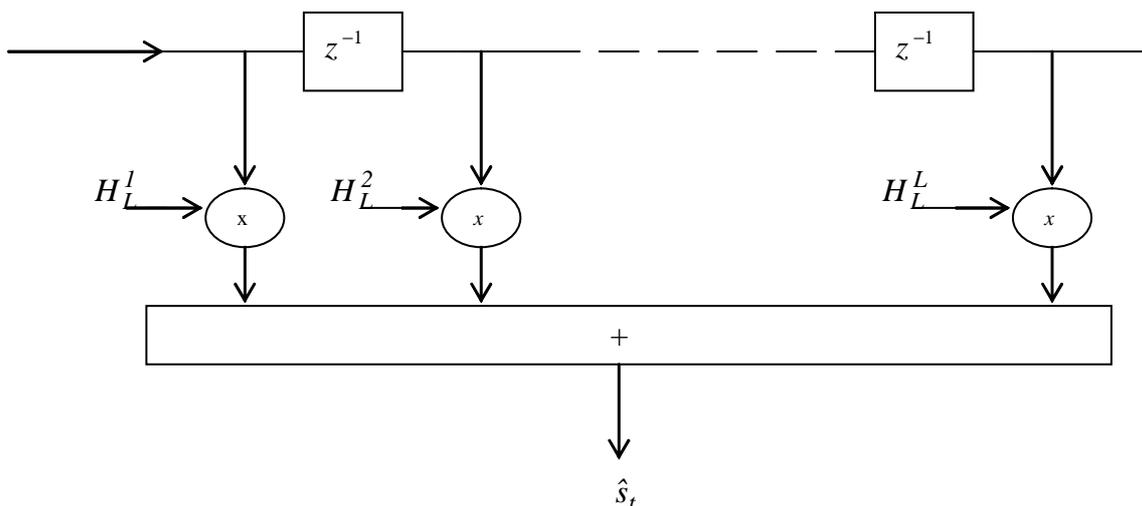


Figure I.4 : Schéma du modèle de filtre transverse

avec

$$\hat{s}_t = \sum_{i=1}^L H_L^i x_{t-i+1}$$

tous les algorithmes présentés dans ce chapitre sont de modèle FIR(MA) permettant d'identifier un canal acoustique. Nous ne parlerons pas de la modélisation ARMA qui n'est pas adaptée au phénomène physique de l'écho, qui ne donne pas de résultats probants. Parmi les algorithmes connus pour l'identification des modèles MA, nous distinguons deux familles principales: la famille des gradients (LMS:Least Mean Square) et celle des moindres carrés récurrents (MCR,) ou bien FTF (Fast Transversal Filter), bien que ces deux algorithmes sont issues de l'algorithme de Newton stochastique. Ces algorithmes seront présentés en détails dans les paragraphes suivants:

I.3. Critères de comparaison entre les algorithmes adaptatifs

Dans la comparaison des performances entre les algorithmes d'identification et de filtrage adaptatifs, plusieurs critères de performances peuvent être utilisés pour évaluer objectivement les performances des nouvelles versions algorithmiques proposées dans la littérature est celle déjà publiées. On distingue deux sortes de critères. La première concerne les performances intrinsèques de l'algorithme, c'est-à-dire sa vitesse de convergence, sa capacité de poursuite et sa robustesse au bruit sur la sortie. La seconde vient des contraintes externes de l'application, englobant d'une part la complexité arithmétique (nombre d'opérations et mémoire requise), la stabilité numérique, et le retard de traitement pour les algorithmes par blocs. Ces critères et d'autres sont résumés comme suite:

- a) L'erreur en régime permanent
- b) Stabilité numérique
- c) Vitesse de convergence en phase initiale
- d) Vitesse de convergence en régime permanent
- e) Influence de la taille du système sur les performances des algorithmes
- f) Influence de la parole comme signal d'excitation
- g) Influence du bruit additif sur le signal écho
- h) Complexité de calcul

I.4. Présentation des algorithmes

Nous allons présenter dans cette partie deux familles d'algorithmes : algorithmes du type gradient stochastique (LMS) , algorithmes du type moindres carrés exacts (RLS) et un algorithme des moindres carrés transversaux rapides (FTF) .

I.4.1. Algorithmes du gradient stochastique LMS et NLMS

L'idée des algorithmes de type gradient stochastique est de remplacer la moyenne statistique dans l'algorithme du gradient déterministe d'équation (I.6) par sa valeur instantanée. On obtient la relation suivante :

$$H_{L,t} = H_{L,t-1} + 2\mu \bar{\epsilon}_{L,t} X_{L,t} \quad (I.7)$$

où $\bar{\epsilon}_{L,t}$ est donnée par la relation (I.6ii).

d'après [15], la condition nécessaire et suffisante de convergence de l'algorithme (LMS) est :

$$0 < \mu < 1/\lambda_{max} \quad (I.7i)$$

où λ_{max} est la plus grande valeur de la matrice d'autocorrélation $R_{L,xx}$ donnée par (I.5).

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

$$0 < \mu \leq \frac{1}{\text{Trace}(R_{L,xx})} = \frac{1}{L\sigma_x^2} \quad (\text{I.7ii})$$

l'algorithme du gradient stochastique normalisé NLMS est une variante du LMS dont le gain d'adaptation est normalisé par l'énergie du signal d'entrée x_t ; il est donné par les équations suivantes:

- Filtrage de y_t

$$\bar{e}_{L,t} = y_t - H_{L,t}^T X_{L,t}$$

- Adaptation du filtre

$$H_{L,t} = H_{L,t-1} + \frac{\delta}{X_{L,t}^T X_{L,t}} \bar{e}_{L,t} X_{L,t}$$

la condition suffisante de convergence est :

$$0 < \delta < 2$$

Ses avantages sont : la faible complexité des calculs $3N$ ou $2N$ (mesurée en nombre de multiplication par itération) et la stabilité numérique.

Son principal inconvénient est que le transitoire dépend de la dispersion spectrale du signal d'entrée ; la convergence peut être très lente dans le cas de la parole.

I.4.2. Algorithmes des moindres carrés exacts

I.4.2.1. Algorithme des moindres carrés récursifs (RLS)

Dans la méthode des moindres carrés exacts, on vise à minimiser par rapport au vecteur des paramètres $H_{L,t}$, à chaque instant t , un critère défini sur les erreurs commise depuis l'instant initial. Ce critère est donné par [5] :

$$J_{L,t} = \sum_{i=1}^t \lambda^{t-i} (y_i - H_{L,t}^T X_{L,i})^2 \quad (\text{I.8})$$

où λ ($0 < \lambda \leq 1$) est un facteur d'oubli exponentiel qui permet à l'algorithme d'oublier le passé trop lointain et de poursuivre les non-stationnarités intervenant dans les signaux.

On suppose que les signaux x_t et y_t sont nuls avant l'instant initial $t = 0$; c'est à dire on se place dans le cas de la fenêtre antérieure.

La solution qui exprime la nullité du gradient de la fonctionnelle $J_{L,t}$

$$\nabla_H J_{L,t} = 0 \Leftrightarrow -2 \left[\sum_{i=1}^t \lambda^{t-i} X_{L,i} y_i - \sum_{i=1}^t \lambda^{t-i} X_{L,i} X_{L,i}^T H_{L,t} \right] = 0$$

est donnée par

$$R_{L,t} H_{L,t} = P_{L,t} \quad (\text{I.9})$$

$$H_{L,t} = R_{L,t}^{-1} P_{L,t} \quad (\text{I.10})$$

où la matrice $R_{L,t}$ représente une matrice d'autocorrélation à court terme qui est donnée par la relation suivante

$$R_{L,t} = \sum_{i=1}^t \lambda^{t-i} X_{L,i} X_{L,i}^T \quad (\text{I.11})$$

$$= \lambda R_{L,t-1} + X_{L,t} X_{L,t}^T \quad (\text{I.11i})$$

le vecteur $P_{N,t}$ désigne un vecteur d'intercorrélation à court terme donnée par la relation :

$$P_{L,t} = \sum_{i=1}^t \lambda^{t-i} X_{L,i} y_i \quad (\text{I.12})$$

$$= \lambda P_{L,t-1} + X_{L,t} y_t \quad (\text{I.12i})$$

En substituant les quantités (déduites de (I.11i) et (I.12i))

$$\lambda R_{L,t-1} = R_{L,t} - X_{L,t} X_{L,t}^T$$

$$\lambda P_{L,t-1} = P_{L,t} - X_{L,t} y_t$$

dans la version de (I.9)

$$\lambda R_{L,t-1} H_{L,t-1} = \lambda P_{L,t-1}$$

on obtient la solution équivalente à (I.10) mais sous une forme récursive :

$$H_{L,t} = H_{L,t-1} - C_{L,t} \bar{\epsilon}_{L,t} \quad (\text{I.13})$$

avec

$$C_{L,t} = \begin{bmatrix} C_{L,t}^1 \\ C_{L,t}^2 \\ \cdot \\ \cdot \\ \cdot \\ C_{L,t}^L \end{bmatrix} = -R_{L,t}^{-1} X_{L,t} \quad (\text{I.14})$$

$$\bar{\epsilon}_{L,t} = y_t - H_{L,t-1}^T X_{L,t}$$

où le vecteur $C_{L,t}$ est appelé gain de Kalman .

L'algorithme (I.13) nécessite l'inversion d'une matrice carré d'ordre L dont le coût de calcul est de l'ordre L^3 opérations arithmétiques par itération .

L'algorithme des moindres carrés récurrents (RLS) résoud ce problème, avec un nombre d'opérations arithmétiques proportionnel à L^2 . En appliquant à la matrice $R_{L,t}$ le lemme d'inversion matricielle suivant [5] :

$$(A + UV^T)^{-1} = A^{-1} - \frac{A^{-1}UV^T A^{-1}}{1 + V^T A^{-1}U}$$

avec :

$$A = \lambda R_{L,t-1} \quad ; \quad U = V = X_{L,t}$$

il vient

$$R_{L,t}^{-1} = \lambda^{-1} \left[R_{L,t-1}^{-1} - \frac{R_{L,t-1}^{-1} X_{L,t} X_{L,t}^T R_{L,t-1}^{-1}}{\lambda + X_{L,t}^T R_{L,t-1}^{-1} X_{L,t}} \right] \quad (\text{I.15})$$

Cet algorithme, appelé « algorithme des moindres carrés récurrents (RLS) », permet, en propageant une matrice carré $L \times L$, de trouver itérativement la solution qui minimise le critère (I.9). Cette solution nécessite un coût en opération arithmétique proportionnel à L^2 . Cette dernière décennie, on s'est aperçu que la solution des moindres carrés pouvait être obtenu avec un coût opératoire proportionnel à L ; ceux sont les algorithmes des moindres carrés rapides.

On se limite aux algorithmes des moindres carrés transversaux rapides non normalisés à oubli exponentiel obtenus dans le cas d'un pré-fenêtrage des signaux d'entrée (les signaux sont supposés nuls avant l'instant initial) .

On utilisera souvent la dénomination algorithme des moindres carrés rapides (MCR) pour désigner ce type d'algorithme.

Dans cette catégorie, on génère la solution des moindres carrés, à chaque instant t , pour un ordre donné L en propageant trois vecteurs d'ordre L (deux prédicteurs aller/retour et le gain de Kalman) au lieu d'une matrice .

Les algorithmes des moindres carrés rapides les plus rapide connus appartiennent à cette catégorie.

I.4.2.2. Algorithmes des moindres carrés transversaux rapides (FTF)

Dans ce paragraphe nous décrivons brièvement l'essentiel des équations qui permettent d'obtenir les algorithmes des moindres carrés transversaux rapides à oubli exponentiel pour le cas de la fenêtre antérieure, faisant intervenir des erreurs a posteriori [10] [11].

Le but de cette méthode des moindres carrés est de trouver le vecteur $H_{L,t}$ qui minimise le critère (I.8) . Une solution récursive à ce problème est donnée par l'algorithme (I.13). Cet algorithme utilise une erreur de filtrage a priori et un vecteur gain d'adaptation d'ordre L , qu'on a appelé gain de Kalman (I.14).

Si on substitue (I.11i) et (I.12i) dans (I.9), on obtient une autre forme récursive du vecteur $H_{L,t}$, strictement équivalente à (I.13), faisant intervenir une erreur de filtrage a posteriori et un gain d'adaptation appelé gain de Kalman dual :

$$H_{L,t} = H_{L,t-1} - \tilde{C}_{L,t} \varepsilon_{L,t} \quad (\text{I.16})$$

avec

$$\varepsilon_{L,t} = y_t - H_{L,t}^T X_{L,t} \quad (\text{I.17})$$

$$\tilde{C}_{L,t} = -\lambda^{-1} R_{L,t-1}^{-1} X_{L,t} \quad (\text{I.18})$$

où $\varepsilon_{L,t}$ donnée par l'expression (I.17) désigne l'erreur de filtrage a posteriori (calculée après la mise à jour du filtre) et $\tilde{C}_{L,t}$ donnée par l'expression (I.18) désigne le gain de Kalman dual.

L'algorithme (I.16) ne peut être appliqué tel quel car l'adaptation et le filtrage ne peuvent être réalisés en même temps. Comme nous allons le voir par la suite, l'un des avantages des algorithmes MCR est que les erreurs a posteriori se calcule à partir des erreurs a priori avant l'opération d'adaptation.

Pour cela, on exploite certaines propriétés d'invariance par décalage du vecteur signal d'entrée étendu à l'ordre $L+1$:

$$X_{L,t} = \begin{bmatrix} x_t \\ X_{L,t-1} \end{bmatrix} \quad (\text{I.19i})$$

$$X_{L,t} = \begin{bmatrix} X_{L,t} \\ x_{t-L} \end{bmatrix} \quad (\text{I.19ii})$$

Ces deux formes du vecteur signal permettent de définir deux matrices d'autocorrélation d'ordre $(L+1)$ partitionnées.

La première qui correspond au vecteur (I.19i) s'écrit :

$$\begin{aligned} R_{L+1,t} &= \sum_{i=1}^t \lambda^{t-i} X_{L+1,i} X_{L+1,i}^T = \sum_{i=1}^t \lambda^{t-i} \begin{bmatrix} x_i \\ X_{L,i-1} \end{bmatrix} \begin{bmatrix} x_i & X_{L,i-1}^T \end{bmatrix} \\ &= \begin{bmatrix} P_{0,t}^a & P_{L,t}^{aT} \\ P_{L,t}^a & R_{L,t-1} \end{bmatrix} \end{aligned} \quad (\text{I.20i})$$

avec :

$$\begin{aligned} P_{0,t}^a &= \sum_{i=1}^t \lambda^{t-i} x_i^2 \\ P_{L,t}^a &= \sum_{i=1}^t \lambda^{t-i} x_i X_{L,i} = \lambda P_{L,t-1}^a + x_t X_{L,t-1} \end{aligned}$$

où l'exposant a, désigne une variable aller .

La deuxième, qui correspond au vecteur (I.19ii) s'écrit :

$$\begin{aligned} R_{L+1,t} &= \sum_{i=1}^t \lambda^{t-i} \begin{bmatrix} X_{L,i} \\ x_{i-L} \end{bmatrix} \begin{bmatrix} X_{L,i}^T & x_{i-L} \end{bmatrix} \\ &= \begin{bmatrix} R_{L,t} & P_{L,t}^r \\ P_{L,t}^{rT} & P_{0,t}^r \end{bmatrix} \end{aligned} \quad (\text{I.20ii})$$

avec :

$$\begin{aligned} P_{0,t}^r &= \sum_{i=1}^t \lambda^{t-i} x_{i-L}^2 \\ P_{L,t}^r &= \sum_{i=1}^t \lambda^{t-i} x_{i-L} X_{L,i} = \lambda P_{L,t-1}^r + x_{t-L} X_{L,t} \end{aligned}$$

où l'exposant ^r, désigne une variable retour.

Le but des algorithmes MCR est de propager un vecteur gain. Le calcul du gain de Kalman (I.14) ou du gain de Kalman dual (I.18) fait intervenir l'inverse de la matrice d'autocorrélation à court terme.

Pour réaliser cette inversion, on utilise le lemme d'inversion d'une matrice partitionnée [10]. Si M désigne une matrice partitionnée :

$$M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$$

L'inverse de cette matrice M s'écrit de manière générale (on suppose la compatibilité des dimensions et l'existence des inverses de certaines matrices de M) .

$$M^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & M_4^{-1} \end{bmatrix} + \begin{bmatrix} I \\ -M_4^{-1}M_3 \end{bmatrix} (M_1 - M_2M_4^{-1}M_3)^{-1} [I - M_2M_4^{-1}] \quad (\text{I.21i})$$

ou

$$M^{-1} = \begin{bmatrix} M_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -M_1^{-1}M_2 \\ I \end{bmatrix} (M_4 - M_3M_1^{-1}M_2)^{-1} [-M_3.M_1^{-1} \ I] \quad (\text{I.21ii})$$

En utilisant la forme (I.21i) pour inverser la matrice (I.20i) et la forme (I.21ii) pour inverser la matrice (I.20ii), on aura les expressions suivantes :

$$R_{L,t}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & R_{L,t-1}^{-1} \end{bmatrix} + \begin{bmatrix} I \\ -R_{L,t-1}^{-1}P_{L,t}^a \end{bmatrix} (P_{0,t}^a - P_{L,t}^{aT} R_{L,t-1}^{-1} P_{L,t}^a)^{-1} [I - P_{L,t}^{aT} R_{L,t-1}^{-1}] \quad (\text{I.22i})$$

$$R_{L,t}^{-1} = \begin{bmatrix} R_{L,t}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -R_{L,t}^{-1}P_{L,t}^r \\ I \end{bmatrix} (P_{0,t}^r - P_{L,t}^{rT} R_{L,t}^{-1} P_{L,t}^r)^{-1} [-P_{L,t}^{rT} R_{L,t}^{-1} \ I] \quad (\text{I.22ii})$$

La forme (I.22i) fait apparaître un prédictor aller optimal au sens des moindres carrés (le terme aller désigne, la modélisation de l'échantillon x_t par une combinaison linéaire de son passé) :

$$a_{L,t} = R_{L,t-1}^{-1} P_{L,t}^a \quad (\text{I.23})$$

Ce vecteur peut être obtenu en minimisant le critère des moindres carrés suivants :

$$J_{L,t}^a = \sum_{i=1}^t \lambda^{t-i} (x_i - a_{L,t}^T X_{L,i-1})^2 \quad (\text{I.24})$$

Les versions récursives du prédictor aller (I.23) faisant intervenir soit le gain de Kalman (I.14), soit le gain de Kalman dual (I.18), s'obtiennent de la même façon que les versions récursives (I.13) et (I.16) du filtre transverse $H_{N,t}$:

$$a_{L,t} = a_{L,t-1} - C_{L,t-1} \bar{e}_{L,t} \quad (\text{I. 25i})$$

$$a_{L,t} = a_{L,t-1} - \tilde{C}_{L,t-1} e_{L,t} \quad (\text{I.25ii})$$

où $\bar{e}_{L,t}$ et $e_{L,t}$ désignent respectivement l'erreur de prédiction aller a priori et l'erreur de prédiction aller a posteriori, qui sont données par :

$$\bar{e}_{L,t} = x_t - a_{L,t-1}^T X_{L,t-1} \quad (\text{I.26i})$$

$$e_{L,t} = x_t - a_{L,t}^T X_{L,t-1} \quad (\text{I.26ii})$$

De la même façon, la forme (I.22ii) fait apparaître un prédicteur retour optimal au sens des moindres carrés (le terme retour désigne la modélisation de l'échantillon x_{t-L} par une combinaison linéaire des échantillons futurs) :

$$b_{L,t} = R_{L,t}^{-1} P_{L,t}^r \quad (\text{I.27})$$

Ce vecteur peut être obtenu en minimisant le critère des moindres carrés suivant :

$$J_{L,t}^r = \sum_{i=1}^t \lambda^{t-i} (x_{i-L} - b_{L,t}^T X_{L,i})^2 \quad (\text{I.28})$$

Les versions récursives de $b_{L,t}$, s'obtiennent en substituant dans la solution (I.27) des versions récursives de $R_{L,t}$ et de $P_{L,t}^r$. Elles s'écrivent :

$$b_{L,t} = b_{L,t-1} - C_{L,t} \bar{r}_{L,t} \quad (\text{I.29i})$$

$$\tilde{b}_{L,t} = b_{L,t-1} - \tilde{C}_{L,t} r_{L,t} \quad (\text{I.29ii})$$

où $\bar{r}_{L,t}$ et $r_{L,t}$ désignent respectivement l'erreur de prédiction retour a priori et l'erreur de prédiction retour a posteriori, qui sont données par :

$$\bar{r}_{L,t} = x_{t-L} - b_{L,t-1}^T X_{L,t} \quad (\text{I.30i})$$

$$r_{L,t} = x_{t-L} - b_{L,t}^T X_{L,t} \quad (\text{I.30ii})$$

Les termes entre parenthèses dans les expressions (I.22i) et (I.22ii) représentent respectivement les variances des erreurs de prédiction aller et retour (minima des critères (I.24) et (I.28)).

Leurs versions récursives s'écrivent [10] :

$$\alpha_{L,t} = \lambda \alpha_{L,t-1} + \bar{e}_{L,t} e_{L,t} \quad (\text{I.31})$$

$$\beta_{L,t} = \lambda \beta_{L,t-1} + \bar{r}_{L,t} r_{L,t} \quad (\text{I.32})$$

où $\alpha_{L,t}$ désigne la variance des erreurs de prédiction aller et $\beta_{L,t}$ désigne la variance des erreurs de prédiction retour .

Les équations qui vont permettre de propager les gains de KALMAN d'ordre $(L+1)$ s'obtiennent en multipliant à droite les expressions (I.22i) et (I.22ii) par le vecteur $[-X_{L+1,t}]$ convenablement partitionné :

$$C_{L+1,t} = \begin{bmatrix} 0 \\ C_{L,t-1} \end{bmatrix} - \frac{e_{L,t}}{\alpha_{L,t}} \begin{bmatrix} 1 \\ -a_{L,t} \end{bmatrix} \quad (\text{I.33i})$$

$$C_{L+1,t} = \begin{bmatrix} C_{L,t} \\ 0 \end{bmatrix} - \frac{r_{L,t}}{\beta_{L,t}} \begin{bmatrix} -b_{L,t} \\ 1 \end{bmatrix} \quad (\text{I.33ii})$$

De la même façon en multipliant à droite les expressions (I.22i) et (I.22ii) prises aux instants $t-1$ par $[-\lambda^{-1} X_{L+1,t}]$ convenablement partitionné, on obtient les deux expressions du gain de KALMAN dual d'ordre $L+1$:

$$\tilde{C}_{L+1,t} = \begin{bmatrix} 0 \\ \tilde{C}_{L,t-1} \end{bmatrix} - \frac{\bar{e}_{L,t}}{\lambda \alpha_{L,t-1}} \begin{bmatrix} 1 \\ -a_{L,t-1} \end{bmatrix} \quad (\text{I.34i})$$

$$\tilde{C}_{L+1,t} = \begin{bmatrix} \tilde{C}_{L,t} \\ 0 \end{bmatrix} - \frac{\bar{r}_{L,t}}{\lambda \beta_{L,t-1}} \begin{bmatrix} -b_{L,t-1} \\ 1 \end{bmatrix} \quad (\text{I.34ii})$$

Il faut noter que les expressions (I.33) et (I.34) font apparaître des erreurs de prédiction qui peuvent être calculées, sans les relations de convolutions (I.26) et (I.30), si on dispose de la première ou de la $(N+1)$ ième composante du gain de Kalman ou du gain de Kalman dual :

$$e_{L,t} = -\alpha_{L,t} C_{L+1,t}^1 \quad (\text{I.35i})$$

$$\bar{e}_{L,t} = -\lambda \alpha_{L,t-1} \tilde{C}_{L+1,t}^1 \quad (\text{I.35ii})$$

$$r_{L,t} = -\beta_{L,t} C_{L+1,t}^{L+1} \quad (\text{I.36i})$$

$$\bar{r}_{L,t} = -\lambda \beta_{L,t-1} \tilde{C}_{L+1,t}^{L+1} \quad (\text{I.36ii})$$

Par ailleurs, parmi les relations qui permettent de réduire la complexité dans les algorithmes, on trouve les relations qui lient les erreurs a priori aux erreurs a posteriori. Pour voir cela, il suffit de prendre par exemple l'erreur de filtrage a posteriori (I.17) et remplacer le filtre $H_{N,t}$ par son équation d'adaptation (I.13) il vient :

$$\varepsilon_{L,t} = \bar{\varepsilon}_{L,t} \left(I + C_{L,t}^T X_{L,t} \right) \quad (\text{I.37})$$

Le terme entre parenthèse dans cette expression définit ce que l'on appelle la variable de vraisemblance

$$\gamma_{L,t} = I + C_{L,t}^T X_{L,t} = I - X_{L,t}^T R_{L,t}^{-1} X_{L,t} \quad (\text{I.38})$$

En théorie, la matrice $R_{L,t}$ et son inverse sont définis positifs . La variable $\gamma_{L,t}$ sera donc comprise entre 0 et 1 pour un fonctionnement normal de l'algorithme .

De la même façon, on obtient les relations entre les erreurs de prédiction a priori et a posteriori

$$e_{L,t} = \gamma_{L,t-1} \bar{e}_{L,t} \quad (\text{I.39})$$

$$\bar{r}_{L,t} = \gamma_{L,t} \bar{r}_{L,t} \quad (\text{I.40})$$

D'autre part, les équations de mise à jour des vecteurs $H_{L,t}$, $a_{L,t}$ et $b_{L,t}$ utilisant soit les erreurs a priori, soit les erreurs a posteriori, sont strictement équivalentes.

Ceci conduit, en égalant deux d'entre elles, à la relation qui lie le gain de Kalman et le gain de Kalman dual :

$$C_{L,t} = \gamma_{L,t} \tilde{C}_{L,t} \quad (\text{I.41})$$

En substituant cette relation dans (I.38) on obtient la relation suivante :

$$\gamma_{L,t} = \frac{1}{1 - \tilde{C}_{L,t}^T X_{L,t}} \quad (\text{I.42})$$

Enfin, au lieu d'utiliser les définitions (I.38) ou (I.42) de la variable de vraisemblance, on utilise des versions plus rapides pour réduire la complexité. ceci s'obtient en appliquant à la définition (I.38) les formes (I.33).

En multipliant (I.33i) à gauche par le vecteur $[x_t \ X_{L,t-1}^T]$, on trouve l'expression de la variable de vraisemblance d'ordre $L+1$:

$$\gamma_{L+1,t} = \frac{\lambda \alpha_{L,t-1}}{\alpha_{L,t}} \gamma_{L,t-1} \quad (\text{I.43})$$

De la même manière, en multipliant (I.33ii) à gauche par le vecteur $[X_{L,t}^T \ x_{t-L}]$ il vient:

$$\gamma_{L+1,t} = \frac{\lambda \beta_{L,t-1}}{\beta_{L,t}} \gamma_{L,t} \quad (\text{I.44})$$

et si on pose que :

$$\theta_{L,t} = \frac{\lambda \beta_{L,t-1}}{\beta_{L,t}} \quad (\text{I.45})$$

et en manipulant les expressions (I.32) et (I.36) on aura d'autre formes d'écritures

$$\begin{aligned} \theta_{L,t} &= 1 + \bar{r}_{L,t} C_{L+1,t}^{L+1} \\ &= 1 + \bar{r}_{L,t} \gamma_{L+1,t} \tilde{C}_{L+1,t}^{L+1} \end{aligned} \quad (\text{I.45i})$$

On calcule alors la variable de vraisemblance d'ordre L par la relation suivante :

$$\gamma_{L,t} = \frac{\gamma_{L+1,t}}{\theta_{L,t}} \quad (\text{I.46})$$

On montre aussi, en utilisant l'expression (I.15), la définition de la variable de vraisemblance (I.38), la propriété $\det[I+XY] = \det[I+YX]$ (det: déterminant d'une matrice) et les liens entre les déterminants des matrices $R_{N+1,t}$ et $R_{N,t}$ que la variable de vraisemblance peut se mettre sous la forme suivante [10] :

$$\gamma_{L,t} = \lambda^L \frac{\det[R_{L,t-1}]}{\det[R_{L,t}]} = \lambda^L \frac{\beta_{L,t}}{\alpha_{L,t}} \quad (\text{I.47})$$

I.5. Algorithme FTF 7L

A partir du paragraphe précédent (I.5.2.2), un grand nombre d'algorithmes peuvent être obtenus. Tous ces algorithmes sont théoriquement équivalents et calculent la solution des moindres carrés avec un nombre de multiplications proportionnelle à l'ordre L .

Les algorithmes des MCR les plus rapides se caractérisent par le gain de KALMAN dual et font intervenir les erreurs a posteriori calculées de manière simple à partir des erreurs a priori. Leur complexité arithmétique est de l'ordre de $7L$ multiplications par échantillons.

Une version a été obtenue par J.M.Cioffi et Kailath [11] ; cet algorithme propage la variable de vraisemblance $\gamma_{N,t}$, appelé FTF "Fast Transversal Filter". L'algorithme *FTF 7L* ($7L$ pour la complexité) est résumé dans la table I.1.

- Variables disponibles à l'instant t :
$a_{L,t-1}; b_{L,t-1}; \tilde{C}_{L,t-1}; \gamma_{L,t-1}; \alpha_{L,t-1}; \beta_{L,t-1}; H_{L,t-1}$
- Nouvelles informations : x_t et x_{t-L}
- Modélisation de x_t et x_{t-L} $\bar{e}_{L,t} = x_t - a_{L,t-1}^T X_{L,t-1}$ $\alpha_{L,t} = \lambda \alpha_{L,t-1} + \gamma_{L,t-1} \bar{e}_{L,t}^2$ $\gamma_{L+1,t} = \frac{\lambda \alpha_{L,t-1}}{\alpha_{L,t}} \gamma_{L,t-1}$ $\tilde{C}_{L+1,t} = \begin{bmatrix} 0 \\ \tilde{C}_{L,t-1} \end{bmatrix} - \frac{\bar{e}_{L,t}}{\lambda \alpha_{L,t-1}} \begin{bmatrix} 1 \\ -a_{L,t-1} \end{bmatrix}$ $a_{L,t} = a_{L,t-1} - \bar{e}_{L,t} \gamma_{L,t-1} \tilde{C}_{L,t-1}$ $\bar{r}_{L,t} = -\lambda \beta_{L,t-1} \tilde{C}_{L+1,t}^{L+1}$ $\gamma_{L,t} = \frac{\gamma_{L+1,t}}{1 + \gamma_{L+1,t} \bar{r}_{L,t} \tilde{C}_{L+1,t}^{L+1}}$ $\begin{bmatrix} \tilde{C}_{L,t} \\ 0 \end{bmatrix} = \tilde{C}_{L+1,t} - \tilde{C}_{L+1,t}^{L+1} \begin{bmatrix} -b_{L,t-1} \\ 1 \end{bmatrix}$ $b_{L,t} = b_{L,t-1} - \bar{r}_{L,t} \gamma_{L,t} \tilde{C}_{L,t}$ $\beta_{L,t} = \lambda \beta_{L,t-1} + \gamma_{L,t} \bar{r}_{L,t}^2$
- Filtrage de y_t
$\bar{\epsilon}_{L,t} = y_t - H_{L,t-1}^T X_{L,t}$ $H_{L,t} = H_{L,t-1} - \bar{\epsilon}_{L,t} \gamma_{L,t} \tilde{C}_{L,t}$

Table I.1 : algorithme *FTF 7L*

I.6. Stabilisation numérique de l'algorithme FTF7L

Il est bien connu que l'efficacité en complexité de calcul des algorithmes des moindres carrés transversaux rapides (FTF) est payée par une dégradation importante de leurs propriétés numériques.

Les erreurs numériques se propagent d'une manière non bornée au cours du temps, pour un facteur d'oubli inférieur à 1, ce qui conduit à une solution instable à plus au moins long terme. Des efforts importants ont été faits pour expliquer l'origine de cette instabilité numérique.

Plusieurs méthodes ont été proposées pour combattre le problème de l'instabilité numérique [6],[13].

En ce qui concerne l'algorithme *FTF7L*, nous avons utilisé la version de stabilisation de base qui a été proposé en [7].

Cette dernière qui est basée sur la rétroaction des erreurs numériques dans le calcul des variables de prédiction rétrogrades est donnée par les équations suivantes :

La variable « indicateur de divergence $\xi_{L,t}$ » est choisi et donnée par la relation suivante :

$$\xi_{L,t} = \bar{r}_{L,t}^c - \bar{r}_{L,t}^{f0} \quad (\text{I.48})$$

avec

$$\bar{r}_{L,t}^c = x_{t-L} - b_{L,t-1}^T X_{L,t} \quad (\text{I.49})$$

$$\bar{r}_{L,t}^{f0} = -\lambda \beta_{L,t-1} \tilde{C}_{L+1,t}^{L+1} \quad (\text{I.50})$$

$$\bar{r}_{L,t}^{f1} = -\lambda^{-L+1} \gamma_{L,t-1} \alpha_{L,t-1} \tilde{C}_{L+1,t}^{L+1} \quad (\text{I.51})$$

pour assurer la stabilité de l'algorithme, il faut satisfaire la condition suivante [7].

$$\lambda > 1 - \frac{1}{2L + 3.5} \quad (\text{I.52})$$

I.7. Initialisation de l'algorithme FTF 7L

Dans le cas de la fenêtre antérieure, les algorithmes MCR supposent que les signaux X_t et y_t sont nuls avant l'instant initial $t=0$. Par conséquent, les erreurs de prédiction retour et le prédicteur retour doivent être nuls avant l'instant $t=L$.

Citons ici la procédure d'initialisation la plus simple et la plus sûre du point de vue numérique pour $t = 0$,

$$a_{L,t} = 0_L^\downarrow ; \tilde{C}_{L,t} = 0_L^\downarrow ; \bar{e}_{L,t} = 0 ; \alpha_{L,t} = E_0 \lambda^L ; \beta_{L,t} = E_0 ; \gamma_{L,t} = 1 ; H_{L,t} = H_{L,0}$$

$H_{L,0}$: vecteur arbitraire

pour $t < L$, $b_{L,t} = 0_L^\downarrow; \bar{r}_{L,t} = 0$

La constante E_0 (strictement positive) est la seule qui doit être convenablement choisie. Pendant les premières itérations, les valeurs prises par les variables internes de l'algorithme sont étroitement liées au choix de la constante E_0 .

En pratique, il faut assurer le démarrage de l'algorithme on pourra choisir par exemple la constante E_0 qui vérifie l'inégalité suivante [11]: $E_0 \geq \frac{L}{100} \sigma_x^2$; où σ_x^2 est l'énergie du signal x_t . Des valeurs de E_0 assurant le bon fonctionnement initial sont données par [8].

I.8. Les algorithmes par bloc

L'approche par bloc a été initialement motivée par une volonté de réduction de charge de calcul grâce aux techniques dites de « convolutions rapides ». Pour appliquer ces méthodes, le filtre demeure non adapté pendant la durée d'un bloc d'échantillon. En plus de la réduction de complexité, les techniques fréquentielles améliorent aussi la vitesse de convergence grâce à la priorité d'orthogonalisation de la transformée de Fourier. Les algorithmes que nous appellerons classiques, traitant la réponse impulsionnelle d'un seul tenant ont pour principal défaut de manipuler une taille de bloc importante, liée à la taille du filtre à identifier. Cette contrainte engendre d'une part, un retard de traitement conséquent qui peut être rédhibitoire pour certaines applications, d'autre part une capacité de poursuite faible car le filtre est actualisé à une cadence bloc.

I.8.1. Les nouveaux algorithmes par blocs

Ces algorithmes ont toujours pour but de diminuer la charge de calcul par rapport au *LMS*, d'améliorer leur vitesse de convergence (déjà obtenue avec les algorithmes fréquentiels classiques), mais surtout, de réduire le délai de traitement et de conserver une capacité de poursuite au moins équivalente à celle du *LMS*. Pour cela, l'idée directrice consiste à travailler avec des petits blocs indépendants de la taille du filtre en le décomposant en M sous-filtres de taille N tel que $L=MN$. Malgré la réduction de la taille du bloc dans les algorithmes MDF (Multidelay bloc frequency Domain adaptive Filter), leur capacité de poursuite demeure inférieure à celle du *LMS*.

I.8.2. Séparation entre la taille du filtre et la taille du bloc

Ces algorithmes obtenus parallèlement dans [18] et [19], découplent la taille du bloc L de la taille du filtre. Nous en présentons ici le principe.

Décomposons H_L en le segmentation en M sous-filtre H_L de taille L retardés de $(i-1)L$ échantillons dans le temps:

$$H_L^i = [h_{L_i} \ h_{L_{i+1}} \ \dots \ h_{L_{i+L-2}} \ h_{L_{i+L-1}}]^T \quad (\text{I-53})$$

Pour $i = 0 \dots M-1$. D'après le principe de superposition, le bloc de sortie de taille L , s'exprime comme la somme de M blocs y_i , correspondant à la sortie du filtre H^i . L'équation de filtrage (I-1) se réécrit comme:

$$\varepsilon_t = y_t - \hat{y}_t = y_t - \sum_{i=0}^{M-1} \hat{y}_t^i = y_t - \sum_{i=0}^{M-1} X_t^i H_{t-L}^i \quad (\text{I-54})$$

Avec les matrices carrées X_t^i (de dimension $L \times L$) de Toeplitz définies par:

$$\left(X_t^i \right) = \left[X_{L,t-L_i-L+1} \dots X_{L,t-L_i} \right] \quad (\text{I-55})$$

Les équations d'adaptation de chaque sous filtre H^i , s'obtiennent en calculant la corrélation par bloc $\phi_t^i = \left(X_{t-L}^i \right)^T \varepsilon_t$. L'équation du filtre unidimensionnelle, se décompose alors en M équations,

$$H_t^i = H_{t-L}^i + v \phi_t^i \quad (\text{I-56})$$

Cette décomposition permet d'utiliser des FFT de taille $2L$, en appliquant, par exemple la méthode *OLS* abordée en aux M sous filtres. A chaque matrice X_t , on associe son extension circulante de taille $2L \times 2L$, qui projetée dans le domaine de Fourier donne une matrice diagonale définie par :

$$D_t^i = \text{diag} \left(F \left[x_{t-L_i-2L+1} \dots x_{t-L_i-L} x_{t-L_i-L+1} \dots x_{t-L_i} \right] \right) \quad (\text{I-57})$$

En posant :

$$W_t^i = F \begin{bmatrix} H_t^i \\ 0_L \end{bmatrix} \quad (\text{I-58})$$

Les L dernières composantes du vecteur (I-59) sont égales à y_t ,

$$\sum_{i=0}^{M-1} D_t^i W_{t-L}^i = \begin{bmatrix} 0 \\ \hat{y}_t \end{bmatrix} \quad (\text{I-59})$$

En observant la relation temporelle entre les blocs consécutifs sur (I-57), on en déduit facilement que :

$$D_t^i = D_{t-L}^{t-L} \quad (\text{I-60})$$

Aussi, à chaque nouveau bloc, seul $D_t^0 = \text{diag} \left(F \left[x_{t-2L+1} \dots x_{t-L} x_{t-L+1} \dots x_t \right] \right)$ est à calculer car les autres termes le sont par la relation :

$$D_t^i = D_{t-L_i}^0 \quad (\text{I-61})$$

Ce qui implique que (I-59) s'exprime simplement par :

$$\sum_{i=0}^{M-1} D_{t-L_i}^0 W_{t-L}^i \quad (\text{I-62})$$

Quand aux corrélations ϕ_t , elles sont estimées de la manière suivante:

$$\phi_t^i = (D_{t-L}^0)^{conj} F \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix} \quad (\text{I-63})$$

Enfin, les équations (I-56) transcrites en fréquence nous donnent les lois d'adaptation:

$$W_t = W_{t-L} + \nu \Omega(\phi_t^i) \quad (\text{I-64})$$

Une autre approche, basée sur la transformée de Fourier Court-Terme, est donnée dans [25]. Notons que la méthode *OLA* permet aussi de mettre en œuvre ce découpage par petits blocs. La seule différence avec la méthode *OLS* provient de l'utilisation de fenêtre de pondération, permettant d'accentuer la décorrélation dans le domaine fréquentiel [26].

Tout comme les algorithmes fréquentiels classiques, la normalisation du pas d'adaptation sur chaque fréquence permet d'accélérer la convergence. Malgré cela, la capacité de poursuite reste inférieure à celle du LMS.

I.8.3. Application à des algorithmes équivalents au LMS :

Ces algorithmes ont la particularité d'être mathématiquement équivalents au LMS, tout en étant des algorithmes par blocs [17] [22] [23], d'où l'appellation de Fast Exact LMS (FE-LMS). Ils ont donc une meilleure capacité de poursuite que les MDF/UMDF. Par réarrangement des équations du LMS parcourues pendant un bloc de taille N, une correction de l'erreur bloc qui aurait été calculée par un BLMS est réalisée en estimant une matrice triangulaire inférieure. En utilisant les mêmes notations que dans le chapitre précédent cette correction s'exprime comme :

$$\tilde{\varepsilon}_t = (I_L + \mu D_{L,t})^{-1} \varepsilon_t = G_{L,t} \varepsilon_t \quad (\text{I-65})$$

Avec la matrice diagonale inférieure de taille $L \times L$:

$$D_{L,t} = \begin{pmatrix} 0 & 0 & \dots & \dots & 0 \\ D_{1,t-L+2} & 0 & \dots & \dots & 0 \\ D_{2,t-L+3} & D_{1,t-L+3} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ D_{L-1,t} & D_{L-2,t} & \dots & D_{L,t} & 0 \end{pmatrix} \quad (\text{I-66})$$

Où

$$D_{i,t} = X_{L,t} X_{L,t-i} \quad (\text{I-67})$$

La matrice $D_{L,t}$ est la partie triangulaire inférieure de la matrice $X_{L,t}X_{L,t}^T$ estimant la matrice d'autocorrélation d'ordre L . notons que l'équation de correction de l'erreur se résout par substitution et que le coût d'estimation de $G_{L,t}$ est proportionnel à $L \times L$. Le FE-LMS est un premier pas pour améliorer la capacité de poursuite par rapport au MDF. Sa version fréquentielle, le FD-LMS, est identique à celle du MDF en rajoutant la correction de l'erreur. Cependant, pour les petits blocs, la technique de convolution rapide temporelle FAST FIR [27], qui découpe le filtre selon sa décomposition polyphasé, peut être plus économique. Par exemple, pour 3 itérations successives, le FE-LMS avec $L=3$ effectue approximativement $4L$ multiplications au lieu de $6L$ pour le LMS, soit une économie de 33%.

A chaque longueur de filtre, il est possible de trouver une taille de bloc optimale L_{opt} compromis entre la complexité de la partie convolution rapide qui diminue avec L et celle de l'estimation de $G_{L,t}$ qui augmente avec L [28]. Pour $L = 256$, la complexité minimale est obtenue avec $L=32$ pour le FE-LMS, $L=64$ pour le FE-LMS, pour une complexité divisée par rapport au LMS.

On note qu'une amélioration de la poursuite des algorithmes fréquents peut être obtenue par un sur-échantillonnage temporel réalisé par un recouvrement des blocs consécutifs supérieur à L . la cadence de mise à jour du filtre est alors supérieure à la cadence bloc. Cette solution est proposée dans [21] pour la méthode OLA. Il est bien entendu que le sur-échantillonnage engendre un surcoût de complexité. Une autre solution basée sur le FE-LMS, proposée dans [22], consiste à superposer une minimisation d'un nouveau critère bloc $J_B = (1/L)\tilde{\epsilon}_t^T \tilde{\epsilon}_t$ sur les L dernières erreurs commises par le FE-LMS. Cet algorithme, nommé NB-LMS (New Block-LMS), converge plus vite et poursuit mieux que le LMS tout en ayant une complexité moindre [23].

I.9. Conclusion

Dans ce chapitre nous avons présenté le modèle de l'annulation de l'écho acoustique comme étant un problème d'identification adaptative d'un filtre à réponse impulsionnelle finie (RIF) modélisant le chemin d'écho (canal acoustique de couplage entre le haut parleur et le microphone) suivi par l'onde acoustique, ainsi que plusieurs algorithmes de filtrage adaptatifs applicables à l'annulation d'écho acoustique, en particulier l'algorithme du gradient stochastique et les algorithmes des moindres carrés rapides bien connu équivalent au RLS, cet algorithme est numériquement instable. Nous avons utilisé les techniques qui existent dans la littérature pour le stabiliser. Une autre nouvelle famille algorithmique qui a de performances aussi bien équivalentes que celles des algorithmes RLS et FTF et qui s'appelle la famille des algorithmes de filtre transversal de newton rapide (FNTRF, Fast Newton Transversal Filter), cet algorithme sera détaillé dans le chapitre 2 de ce document.

Chapitre II

Présentation des algorithmes des Moindres Carrés Rapides de type Newton

II.1. Introduction

Les algorithmes NLMS et FTF peuvent être considérés comme des versions stochastiques de l'algorithme de Newton qui cherche itérativement la solution H_{opt} de l'équation $\nabla J(t, H(t)) = 0_L$ (0_L est le vecteur nul de taille L). Le critère J minimisé est les moindres carrés moyens $E[\bar{\epsilon}^2(t)]$ dans le cas du NLMS et les moindres carrés pondérés (avec pré-fenêtrage) dans le cas du FTF :

$$J(t) = \sum_{i=1}^t \lambda^{t-i} (y(i) - H^T(t)X(i))^2.$$

dans le cas du NLMS le signal d'entrée $x(t)$ est un bruit blanc (c'est à dire un processeur autorégressif (AR) d'ordre zéro), alors que dans le cas du FTF ce signal est un processeur AR d'ordre L , on notera que cette dernière hypothèse est utilisée explicitement dans les formes transverses rapides des moindres carrés qui mettent en œuvre des prédicteurs d'ordre L . L'idée de base de la nouvelle famille des algorithmes de Fast Newton est que si le signal d'entrée peut être modélisé comme un processus AR d'ordre N inférieur à L , alors on peut construire une approximation optimale de la matrice de covariance $R_L(t)$ d'ordre L par un mécanisme d'extrapolation simple à partir des N premières valeurs de la covariance estimée de l'entrée.

Il en résulte que la mise à jour du gain de Kalman de taille L peut être obtenue à partir des prédicteurs d'ordre N correspondant à un algorithme FTF du même ordre N [29]. Ceci conduit à une importante réduction du volume des calculs lorsque N est très inférieur à L .

La matrice de covariance extrapolée est associée aux prédicteurs aller et retour d'ordre L dont N composantes seulement sont non nulles et égales à celles des prédicteurs associés au problème de prédiction optimale d'ordre N . Les variances des erreurs de prédiction d'ordre L , sont identiques à celles d'ordre N .

II.2. Principe de base de l'algorithme FNTF

Tous les algorithmes des moindres carrés rapides utilisent des prédicteurs pour l'estimation d'un filtre d'ordre L . Dans la plupart des situations, une prédiction d'ordre N inférieur à L est suffisante pour atteindre une connaissance acceptable du signal. La parole est un exemple typique illustrant cette dernière remarque car le gain de prédiction plafonne dès que N dépasse la dizaine.

En remplaçant la taille du filtre L par N dans la partie prédiction des algorithmes des moindres carrés rapides, Il est évident que l'algorithme résultant, s'il converge, ne minimise plus le critère des moindres carrés. Pour concevoir l'existence d'un tel algorithme, il faut se remémorer que le RLS peut aussi s'interpréter comme un algorithme de Newton stochastique, où la Hessien du critère des moindres carrés moyens est justement la matrice de corrélation d'ordre L .

L'algorithme NLMS est aussi un algorithme de Newton stochastique dans lequel la matrice d'autocorrélation est réduite à une matrice diagonale. Aussi, le signal d'excitation est supposé autorégressif d'ordre 0. Malgré cette hypothèse généralement fautive, le NLMS converge même lorsque le signal corrélé, ce qui implique qu'une modélisation AR (L) du signal n'est pas nécessaire. Dans l'article [29], on définit une nouvelle classe d'algorithmes, appelés Fast Newton transversal Filters (FNMF), modélisant le signal d'excitation par un AR(N) avec N compris entre 0 et L . Grâce à la réduction de l'ordre de prédiction de L à N , la complexité de ces algorithmes chute de $8L$ à $O(N) + 2L$. Cette nouvelle famille d'algorithme permet de trouver une estimée du Hessien moins grossière que dans le NLMS et moins complexes que dans le cas des algorithmes RLS rapide.

Dans ce chapitre, nous présentons cette nouvelle famille d'algorithme en faisant intervenir la théorie de la prédiction linéaire, ensuite on donne la dérivation de l'algorithme FNMF. D'autre version dérive de cet algorithme est présentée dans ce chapitre. La première partie de ce chapitre introduit une méthode d'extrapolation des matrices d'autocorrélation dans le cas non stationnaire qui permet d'obtenir un Hessien d'ordre L à partir de celui d'ordre N à partir des prédicteurs associés à cette matrice extrapolée.

II.3. Principe de base de l'extrapolation de la corrélation

Ce problème a déjà été résolu dans les cas stationnaire pour améliorer les techniques classiques d'estimation spectrale par transformée de fourrier [30]. Nous allons en illustrer le principe dans le cas stationnaire pour ensuite l'appliquer par analogie aux estimations récursives effectuées par les MCR [37].

II.3.1. Modélisation statistique de la corrélation (cas stationnaire) :

Les techniques à haute résolution spectrale réalisent une modélisation AR(N) du processus à partir des $N+1$ premiers coefficients de corrélation estimés. Le terme haute résolution provient du fait que l'estimateur spectral AR(N) extrapole implicitement les coefficients de corrélation mesurés alors qu'ils sont imposés nuls dans les techniques classiques (périodiques,...). Rappelons que dans les cas stationnaire Gaussien, cette méthode possède aussi deux autres interprétations le maximum d'entropie [31] ou la maximisation de la variance de l'erreur de prédiction [32].

a- Détermination des coefficients extrapolés

Supposons que nous ayons à notre disposition les $N+1$ premiers coefficients de corrélation, $\Gamma_{N+1}^{aT} = [r_0, r_1, \dots, r_N]$, d'un processus stationnaire et que nous voulions les extrapoler jusqu'à l'ordre $L+1$, c'est à dire rechercher $\hat{r}_{N+1}, \dots, \hat{r}_L$ tel que la matrice de corrélation extrapolée R_{L+1}^e , soit définie non négative.

La première étape de l'extrapolation consiste à déterminer le modèle $AR(N)$ du processus à partir des données connus. Pour cela, nous pouvons utiliser l'algorithme de Levinson [33] permettant d'estimer ce modèle $AR(N)$ par récurrence sur l'ordre. Supposons que nous ayons estimé le modèle $AR(k-1)$ à partir de Γ_k^a , alors les trois équations suivantes actualisent le prédicteur directe a_k et la variance de l'erreur de prédiction α_k d'ordre k en fonction du prédicteur rétrograde b_{k-1} , de la variance de prédiction d'ordre $k-1$ ainsi que la nouvelle information r_k :

$$a_k = \begin{bmatrix} a_{k-1} \\ 0 \end{bmatrix} + K_k \begin{bmatrix} b_{k-1} \\ -1 \end{bmatrix}$$

Où K_k est appelé coefficient de réflexion défini par,

$$K_k = a_k^{(k)} = - \left(\frac{r_k - b_{k-1}^T \Gamma_{k-1}^r}{\alpha_{k-1}} \right)$$

et,

$$\alpha_k = (1 - K_k^2) \alpha_{k-1}$$

Rappelons que dans le cas stationnaire, $\Gamma_k^r = J_k \Gamma_k^a$ avec J_k la matrice coidentité de taille $k \times k$, de sorte que $b_k = J_k a_k$. Prenons maintenant $k = N$ et annulons les coefficients de réflexion k_k pour $k > N$, afin de modéliser le processus par $AR(N)$. En annulant K_k pour $N < k \leq L$, on détermine alors les coefficients extrapolés par,

$$r_k^e = \sum_{i=1}^N a_N^{(i)} r_{k-i}^e$$

Pour $k > N$ et $r_k^e = r_k$ pour $0 \leq k \leq N$.

b- Preuve de caractère défini non négatif de la matrice extrapolée

Il faut ensuite prouver le caractère défini non négatif de la matrice R_{L+1}^e . Pour cette démonstration, nous avons utilisé le théorème de Sylvester qui montre qu'une matrice est définie non négative si et seulement si tous les déterminants de ses mineurs principaux, y compris celui de cette matrice, sont strictement positifs.

d'ordre inférieur. Nous allons démontrer que R_{L+1}^e est définie non négative par récurrence. Exprimons premièrement $\det[R_{L+1}^e]$ en fonction de $\det[R_k]$ grâce au lemme suivant concernant le déterminant d'une matrice partitionnée,

$$\det \begin{bmatrix} X & Y \\ Z & W \end{bmatrix} = \det[W] \det[X - YW^{-1}Z] = \det[X] \det[W - ZX^{-1}Y]$$

Appliquons la première partie du lemme à la matrice R_{k+1}^e partitionnée en,

$$R_{k+1}^e = \begin{bmatrix} r_0 & \Gamma_k^{aT} \\ \Gamma_k^a & R_k \end{bmatrix}$$

en posant $X = r_0$, $Y = \Gamma_k^{aT} \begin{bmatrix} \Gamma_{k-1}^{aT} & r_k^e \end{bmatrix}$ et $W = R_k$, il vient,

$$\det[R_{k+1}^e] = \det[R_k] \left(r_0 - \begin{bmatrix} \Gamma_{k-1}^{aT} & r_k^e \end{bmatrix} R_k^{-1} \begin{bmatrix} \Gamma_{k-1}^a \\ r_k^e \end{bmatrix} \right)$$

Appliquons ensuite le lemme d'inversion (I-22i) à R_k , puis injectons l'expression obtenue dans (I-38). Après quelques développements, l'équation (I.38) est réduite à,

$$\det[R_{k+1}^e] = \det[R_k] (\alpha_{k-1})$$

D'après le théorème de Sylvester et la relation précédente, la matrice R_{k+1}^e est donc définie non négative si R_k l'est. En itérant cette démonstration de N à l'ordre L, on démontre que R_{L+1}^e est définie non négative.

Pour trouver le prédicteurs d'ordre L associé à la matrice extrapolée R_{k+1}^e , on annule les coefficients de réflexion pour $K > N$ vient d'après (I-34i) et I-42) que,

$$\alpha_L = \alpha_N$$

et

$$a_L = \begin{bmatrix} a_N \\ 0_{L-N} \end{bmatrix}$$

II.3.2. modélisation réursive (Cas non stationnaire)

La démonstration faite dans ce paragraphe s'inspire du cas "multichannel" présenté dans [34]. Le cas non stationnaire se particularise par ses deux prédicteurs directe et rétrogrades différents. Aussi, l'extrapolation selon un ou l'autre des prédicteurs peut à priori donner deux valeurs différentes pour $r_{k,t}^e$.

a- Détermination des coefficients extrapolés:

Supposons que notre signal d'excitation soit autorégressif d'ordre $k-1$, et que nous ayons déterminé les couples $(a_{k-1,t}, \alpha_{k-1,t})$ et $(b_{k-1,t}, \beta_{k-1,t})$ race à $R_{k,t}$. nous allons voir dans ce qui suit comment déterminer R_{k+1}^e (r_{k+1}) définie non négative et les prédicteurs d'ordre k, à partir des données.

La première étape consiste à mettre en évidence l'inconnue ou plutôt les inconnues en fonction des données connues. Cela peut être réalisé grâce aux deux partitionnements de $R_{k+1,t} (r_{k,t})$ donnés par (I-22i) et (I-22ii).il vient,

$$R_{k+1,t}(\hat{r}_{k,t}^a) = \begin{bmatrix} r_{0,t} & \Gamma_{k,t}^{aT} \\ \Gamma_{k,t}^a & R_{k,t-1} \end{bmatrix}$$

Où

$$R_{k+1,t}(\hat{r}_{k,t}^r) = \begin{bmatrix} R_{k,t} & \Gamma_{k,t}^{rT} \\ \Gamma_{k,t}^r & r_{0,t-k} \end{bmatrix}$$

Tous les coefficients de corrélation $r_{i,t}$ sont estimés par un calcul récursive à oubli exponentiel sauf pour $i= k$. les vecteurs d'intercorrélation s'expriment donc par:

$$\Gamma_{k,t}^a = [r_{1,t}, r_{2,t}, \dots, r_{k-2,t}, r_{k-1,t}, \hat{r}_{L,t}^a]^T = [\Gamma_{k-1,t}^a \hat{r}_{k,t}^a]^T$$

Où $\hat{r}_{k,t}^r$ sera estimé selon la prédiction directe,

$$\Gamma_{k,t}^r = [\hat{r}_{k,t}^b, r_{k-1,t-1}, \dots, r_{2,t-k}, r_{1,t-L+1}]^T = [\hat{r}_{k,t}^r \Gamma_{k-1,t-1}^r]^T$$

Avec $\hat{r}_{k,t}^r$ estimé par la prédiction rétrograde.

Exprimons maintenant le prédicteur directe d'ordre L ainsi que la variance de l'erreur de prédiction en fonction en fonction de l'ordre L-1. Pour cette étape, nous allons faire appel aux équations de base des algorithmes en treillis basés sur la récurrence sur l'ordre [33]. Il vient facilement en manipulant (I-23) et (I-22ii) que,

$$a_{k,t} = \begin{bmatrix} a_{k-1,t} \\ 0 \end{bmatrix} + \begin{bmatrix} b_{k-1,t-1} \\ -I \end{bmatrix} k_{k,t}^a$$

Dans laquelle le coefficient de réflexion $K_{k,t}^a$ dépend de l'inconnue $\hat{r}_{k,t}^a$:

$$K_{k,t}^a = -\frac{1}{\beta_{k-1,t-1}} (\hat{r}_{k,t}^a - \Gamma_{k-1,t-1}^{rT} a_{k-1,t})$$

De même la variance définie en (I-25ii) s'exprime comme,

$$a_{k,t} = a_{k-1,t} - \beta_{k-1,t-1} (k_{k,t}^a)^2$$

Par analogie avec le cas stationnaire, nous annulons le coefficient de réflexion directe $K_{k,t}^a$ pour trouver le coefficient extrapolé $r_{k,t}^a$. Il vient :

$$\hat{r}_{k,t}^a = \Gamma_{k-1,t-1}^{b,t} \mathbf{a}_{k-1,t}$$

Suivons maintenant la même démarche pour la prédiction rétrograde définie en (I-29i) et (I-30i). Les équations du prédicteur rectour et sa variance s'écrivent respectivement:

$$\mathbf{b}_{k,t} = \begin{bmatrix} 0 \\ \mathbf{b}_{k-1,t} \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{a}_{k-1,t-1} \end{bmatrix} k_{k,t}^r$$

$$K_{k,t}^r = -\frac{1}{\alpha_{k-1,t-1}} (\hat{r}_{k,t}^r - \Gamma_{k-1,t-1}^{aT} \mathbf{b}_{k-1,t-1})$$

$$\beta_{k,t} = \beta_{k-1,t} - \alpha_{k-1,t} (k_{k,t}^r)^2$$

L'annulation du coefficient de réflexion rétrograde $K_{k,t}^r$ nous procure en second coefficient extrapolé,

$$\hat{r}_{k,t}^r = \Gamma_{k-1,t-1}^{a,t} \mathbf{b}_{k-1,t}$$

Les deux coefficients extrapolés semblent différents. Cependant d'après (I-23),

$$\hat{r}_{k,t}^a = \Gamma_{k-1,t-1}^{rT} \mathbf{a}_{k-1,t} = \Gamma_{k-1,t-1}^{rT} \mathbf{R}_{k-1,t-1}^{-1} \Gamma_{k-1,t-1}^a$$

Prenons alors la transposée de l'égalité précédente $r_{k,t}^{aT} = r_{k,t}^a$ et utilisons (I-29i), il vient:

$$\Gamma_{k-1,t-1}^{aT} \mathbf{R}_{k-1,t-1}^{-1} \Gamma_{k-1,t-1}^r = \Gamma_{k-1,t-1}^{aT} \mathbf{b}_{k-1,t-1} = r_{k,t}^r$$

La dernière relation prouve que les extrapolations selon le prédicteur directe et le prédicteur rétrograde sont identiques et nous désignons par $r_{k,t}^e$ ce coefficient de corrélation unique. Dans le cas contraire, nous aurions deux matrices extrapolées différentes selon l'une ou l'autre des approches ce qui conduirait à une possibilité de mettre en œuvre les FNTF.

Il reste à construire les autres éléments de la matrice $R_{L+1,t}$. Nous venons de trouver $r_{k,t}^e$ grâce à $R_{k+1,t}$. Pour déterminer les autres éléments de la $k+1^{ième}$ diagonale, il suffit d'utiliser successivement tous les autres mineurs principaux d'ordre $k+1$ ($R_{k+1,t-1}$, $R_{k+1,t-2}$, ...). La construction de la $k+2^{ième}$ diagonale s'appuiera sur tous les mineurs d'ordre $k+2$, ect..., jusqu'à la détermination de $r_{L,t}^e$. Il est simple de trouver une relation récurrente pour déterminer tous les coefficients extrapolés $r_{j,t-i}^e$ pour $k \leq j \leq L$ et $0 \leq i \leq L-k$:

$$r_{j,t-1}^e = \Gamma_{j-1,t-i}^{aT} R_{j-1,t-i}^{-1} \Gamma_{j-1,t-i-1}^r$$

b- Preuve du caractère défini non négatif de la matrice extrapolée:

Ayant déterminé $r_{k,t}^e$, il faut maintenant prouver, comme dans le cas stationnaire, que $R_{k+1,t}^e$ est définie positive pour être candidate comme matrice de corrélation. Une expression analogue à (I-42) s'obtient en développant l'expression de $R_{k+1,t}$:

$$\det[R_{k+1}^e] = \det \left[R_{k,t-1} \begin{pmatrix} r_{0,t} - [\Gamma_{k-1,t}^{aT} r_{k,t}^e] R_{k,t-1}^{-1} [\Gamma_{k-1,t}^a] \\ r_{k,t}^e \end{pmatrix} \right]$$

Puis en exprimant $R_{k,t-1}^{-1}$ sous la forme (I-22i).

$$\det[R_{k+1}^e] = \det[R_{k,t-1}] \alpha_{k-1,t-1}$$

Une relation similaire se trouve en développant l'expression de la matrice $R_{k+1,t}$:

$$\det[R_{k+1}^e] = \det[R_{k,t}] \beta_{k-1,t-1}$$

La matrice $R_{k+1,t}^e$ est donc définie non négative si $R_{k,t}$ ou $R_{k,t-1}$ l'est. Comme cette démonstration est applicable à tous les mineurs principaux de $R_{L+1,t}^e$, elle est donc définie non négative d'après le théorème de Sylvester et elle possède par conséquent toutes les bonnes propriétés pour représenter une matrice d'autocorrélation.

- Prédicteurs associés à la matrice extrapolée

Pour déterminer les prédicteurs associés à $R_{k+1,t}^e$, on réinjecte le coefficient extrapolé $\hat{r}_{k,t}^e$ dans les expressions (I-25i) et (I-31) du prédicteur direct d'ordre k et de la variance correspondante,

$$\mathbf{a}_{k,t} = \begin{bmatrix} \mathbf{a}_{k-1,t} \\ 0 \end{bmatrix}$$

$$\alpha_{k,t} = \alpha_{k-1,t}$$

Puis dans les expressions (I-29i) et (I-32) du prédicteur rétrograde et de la variance associée,

$$\mathbf{b}_{L,t} = \begin{bmatrix} 0 \\ \mathbf{b}_{k-1,t-1} \end{bmatrix}$$

$$\beta_{k,t} = \beta_{k-1,t-1}$$

Remarquons ici un point important. Les prédictions directe et rétrograde travaillent sur des données décalées dans le temps.

Si maintenant nous voulons trouver les prédicteurs d'ordre L associées à $R_{L+1,t}^e$, extrapolée à partir des prédicteurs d'ordre N ($k = N$), nous devons appliquer récursivement de L à N les 4 équations précédentes. Il vient,

$$\mathbf{a}_{L,t} = \begin{bmatrix} \mathbf{a}_{L-1,t} \\ 0 \end{bmatrix}$$

$$\alpha_{L,t} = \alpha_{L-1,t}$$

Puis,

$$\mathbf{b}_{L,t} = \begin{bmatrix} 0 \\ \mathbf{b}_{L-1,t-1} \end{bmatrix}$$

$$\beta_{L,t} = \beta_{L-1,t-1}$$

L'échantillon x_{t-L} est prédicteur rétrograde d'ordre L à l'instant t équivalent au prédicteur rétrograde d'ordre N à l'instant $t-L+N$.

On note que la construction du Hessien $R_{L,t}^e$ de la nouvelle gamme est complexe. Cependant, la mise à jour récursive de son inverse est réalisée comme dans les MCRR, c'est-à-dire par le biais du gain de Kalman extrapolé $\tilde{C}_{L,t} = (R_{L,t}^e)^{-1} X_{L,t}$, en faisant appel aux prédicteurs de la matrice extrapolée qui possèdent la propriété fondamentale d'être tronqués à l'ordre N , ce qui va permettre de réduire la complexité.

Dans le paragraphe suivant, nous pressentons les équations et la dérivation de la version originale de l'algorithme FNTF12N2L.

II.4. Dérivation de la version originale de l'algorithme adaptatif FNTF12N2L

Le modèle standard pour le trajet d'écho acoustique est un filtre transversal avec un grand nombre de coefficients L (plusieurs milliers). Ce modèle *FIR* représente la réponse impulsionnelle $\mathbf{h}_{L,t}$ de taille L du chemin de l'écho (exactement sa première partie, car la réponse impulsionnelle est en générale longue).

Dans cette thèse, on ne considère que les filtres transversaux adaptatifs *FIR*. L'entrée du filtre transversal $\mathbf{h}_{L,t-1}$ est x_t et sa sortie est \hat{y}_t .

L'observation bruitée à la sortie du trajet de l'écho acoustique est y_t . Le filtre adaptatifs $\mathbf{h}_{L,t-1}$ est adapté à chaque nouvel échantillon dans le but de minimiser la puissance de l'erreur de filtrage à la sortie

$$\bar{\epsilon}_{L,t} = y_t - \hat{y}_t.$$

On va faire un rappel de l'équation de la partie de filtrage de l'algorithme FNTF [37,38].

$$\hat{y} = \mathbf{h}_{L,t-1}^T \mathbf{x}_{L,t}, \quad \text{estimation d'écho} \quad (\text{II.1})$$

$$\bar{\varepsilon}_{L,t} = y_t - \hat{y}_t, \quad \text{estimation d'erreur} \quad (\text{II.2})$$

$$\mathbf{h}_{L,t} = \mathbf{h}_{L,t-1} - \tilde{\mathbf{c}}_{L,t} \gamma_{L,t} \bar{\varepsilon}_{L,t}, \quad \text{adaptation du filtre} \quad (\text{II.3})$$

Où T symbolize la transpose d'une matrice. Le vecteurs $\mathbf{x}_{L,t}$ contient les L derniers échantillons d'entrée, $\mathbf{x}_{L,t} = [x_t, x_{t-1}, \dots, x_{t-L+1}]$.

Le gain de Kalman dual $\tilde{\mathbf{c}}_{L,t}$ et la variable de vraisemblance $\gamma_{L,t}$ sont définies comme suite:

$$\tilde{\mathbf{c}}_{L,t} = -\lambda^{-1} \mathbf{R}_{L,t-1}^{-1} \mathbf{x}_{L,t} \quad (\text{II.4})$$

$$\gamma_{L,t} = \frac{1}{1 - \tilde{\mathbf{c}}_{L,t}^T \mathbf{x}_{L,t}} \quad (\text{II.5})$$

Où $\mathbf{R}_{L,t}$ représente la matrice court terme du signal d'entrée, elle est de dimension $L \times L$. Dans le cas des moindres carrés (LS) pondérées exponentiellement, la matrice de covariance est actualiser par l'équation réursive suivante:

$$\mathbf{R}_{L,t} = \lambda \mathbf{R}_{L,t-1} + \mathbf{x}_{L,t} \mathbf{x}_{L,t}^T, \quad 0 < \lambda < 1 \quad (\text{II.7})$$

L'algorithme FTF [29,37] est construit autour d'une estimation réursive efficace de $\tilde{\mathbf{c}}_{L,t}$ et $\gamma_{L,t}$ qui sont dérivés des deux formes équivalentes de la matrice $\mathbf{R}_{L+1,t}^{-1}$

$$\mathbf{R}_{L+1,t}^{-1} = \begin{bmatrix} \mathbf{R}_{L,t}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{b}_{L,t} \\ I \end{bmatrix} \begin{bmatrix} -\mathbf{b}_{L,t}^T & 1 \end{bmatrix} \beta_{L,t}^{-1} \quad (\text{II.8a})$$

$$\mathbf{R}_{L+1,t}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{R}_{L,t-1}^{-1} \end{bmatrix} + \begin{bmatrix} I \\ -\mathbf{a}_{L,t} \end{bmatrix} \begin{bmatrix} 1 & -\mathbf{a}_{L,t}^T \end{bmatrix} \alpha_{L,t}^{-1} \quad (\text{II.8b})$$

Ou $\mathbf{a}_{L,t}$ et $\mathbf{b}_{L,t}$ sont les predicteurs aller et retour d'ordre L , et $\alpha_{L,t}$ et $\beta_{L,t}$ sont les puissance des erreurs de prédiction aller et retour respectivement.

Pour bien illustrer la derivation de l'algorithme FNTEF, on prend un exemple simple, que $N=L-1$. en supposant que le signal d'entrée est autoregressive d'ordre $L-1$, on trouve une matrice $\mathbf{R}_{L+1,t}^a$ définie positive sous deux formes pour bien partitionnées et ceci pour faire apparaître les éléments qu'on va extrapoler. Le résultat est [29]:

$$\mathbf{R}_{L+1,t}^a = \begin{bmatrix} \mathbf{\Gamma}_{0,t}^a & \mathbf{\Gamma}_{L,t-1}^{aT} \\ \mathbf{\Gamma}_{L,t}^a & \mathbf{R}_{L,t-1} \end{bmatrix} \quad (\text{II.9a})$$

$$\mathbf{R}_{L+1,t}^b = \begin{bmatrix} \mathbf{R}_{L,t} & \mathbf{\Gamma}_{L,t-1}^b \\ \mathbf{\Gamma}_{L,t}^{bT} & \mathbf{\Gamma}_{0,t}^b \end{bmatrix} \quad (\text{II.9b})$$

Ou les variances sont:

$$\mathbf{\Gamma}_{0,t}^a = r_{0,t},$$

$$\mathbf{\Gamma}_{0,t}^b = r_{0,t-L}$$

et les vecteurs de corrélation :

$$\mathbf{\Gamma}_{L,t}^a = [r_{1,t}, r_{2,t}, \dots, \hat{r}_{L,t}^a]^T,$$

$$\mathbf{\Gamma}_{L,t}^b = [\hat{r}_{L,t}^b, r_{L-1,t-1}, \dots, r_{1,t-L+1}]^T$$

sont estimés en fonction de $r_{i,t} = \lambda r_{i,t} + x_t x_{t-i}$ for $0 \leq i \leq L-1$.

On note que $\hat{r}_{L,t}^a$ et $\hat{r}_{L,t}^b$ sont les seuls éléments qui sont inconnus. En utilisant la précédente notation, l'optimum au sens des moindres carrés (LS) du prédicteur directe et sa variance d'erreur d'ordre L sont respectivement données comme suite:

$$\mathbf{a}_{L,t} = \mathbf{R}_{L,t-1}^{-1} \mathbf{\Gamma}_{L,t}^a \quad (\text{II.10})$$

$$\alpha_{L,t} = \mathbf{\Gamma}_{0,t}^a - \mathbf{\Gamma}_{L,t}^{aT} \mathbf{a}_{L,t} \quad (\text{II.11})$$

dans l'équation (11), la puissance minimale de l'erreur directe dépend de l'élément inconnu $\hat{r}_{L,t}^a$.

L'idée de base caractérisant l'algorithme FNTF est de calculer cet élément inconnu par la maximisation de $\alpha_{L,t}$. C'est-à-dire, il est choisi pour être le choix le plus critique possible et ceci par rapport l'estimation du prédicteur aller. Ce critère est semblable au principe du maximum d'entropie pour extrapoler l'ordre d'autocorrélation d'un processus stationnaire [31]. Pour réaliser cette maximisation, l'astuce est de tenir en compte de la formulation d'ordre croissant de (10-11) qui peut être obtenue à partir de la théorie du filtre en treillis RLS:

$$\mathbf{a}_{L,t} = \begin{bmatrix} \mathbf{a}_{L-1,t} \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{b}_{L-1,t-1} \\ -I \end{bmatrix} k_{L,t}^a \quad (\text{II.12})$$

$$\alpha_{L,t} = \alpha_{L-1,t} - \beta_{L-1,t-1} (k_{L,t}^a)^2 \quad (\text{II.13})$$

Ou le coefficient de réflexion $k_{L,t}^a$ est définie par:

$$k_{L,t}^a = -\frac{1}{\beta_{L-1,t-1}}(\hat{r}_{L,t}^a - \mathbf{\Gamma}_{L-1,t-1}^{bT} \mathbf{a}_{L-1,t}) \quad (\text{II.14})$$

En effet, la variance de l'erreur retour $\beta_{L-1,t-1}$ est positive et $\alpha_{L,t}$ prend sa valeur maximale lorsque $k_{L,t}^a$ est mise égale à zéro, i.e.

$$\hat{r}_{L,t}^a = \mathbf{\Gamma}_{L-1,t-1}^{bT} \mathbf{a}_{L-1,t} \quad (\text{II.15})$$

En choisissant cette valeur optimale, le prédicteur aller et la puissance de l'erreur minimale du processus extrapolé sont donnée par:

$$\mathbf{a}_{L,t} = \begin{bmatrix} \mathbf{a}_{L-1,t} \\ 0 \end{bmatrix} \quad (\text{II.16})$$

$$\alpha_{L,t} = \alpha_{L-1,t} \quad (\text{II.17})$$

En appliquant la même approche aux paramètres retour, c'est-à-dire minimisation de $\beta_{L,t}$ par rapport $\hat{r}_{L,t}^b$, on trouve $\hat{r}_{L,t}^b = \hat{r}_{L,t}^a$ et :

$$\mathbf{b}_{L,t} = \begin{bmatrix} 0 \\ \mathbf{b}_{L-1,t-1} \end{bmatrix} \quad (\text{II.18})$$

$$\beta_{L,t} = \beta_{L-1,t-1} \quad (\text{II.19})$$

Les résultats montrent que $\mathbf{R}_{L+1,t}^a = \mathbf{R}_{L+1,t}^b$ peuvent être extrapolés à partir de $\mathbf{R}_{L,t}$. Les prédicteurs correspondants sont donnés par (16-17) or (18-19). Ces prédicteurs d'ordre L sont égaux au prédicteurs optimal d'ordre $L-1$ étendue avec un zéro.

Revenant au cas général de l'extrapolation de N à L . en suivant un raisonnement similaire, il devient possible d'extrapoler $\mathbf{R}_{N+1,t}$ à $\mathbf{R}_{L+1,t}$ de manière récursive. En appliquant (16-17) et (18-19) récursivement de L à N , on obtient les expressions suivantes:

$$\mathbf{a}_{L,t} = \begin{bmatrix} \mathbf{a}_{N,t} \\ \mathbf{0}_{L-N} \end{bmatrix} \quad (\text{II.20})$$

$$\alpha_{L,t} = \alpha_{N,t} \quad (\text{II.21})$$

$$\mathbf{b}_{L,t} = \begin{bmatrix} \mathbf{0}_{L-N} \\ \mathbf{b}_{N,t_D} \end{bmatrix} \quad (\text{II.22})$$

$$\beta_{L,t} = \beta_{N,t_D} \quad (\text{II.23})$$

avec $t_D = t - L + N$. en utilisant ces prédicteurs tronqué à la place des prédicteur d'ordre complet comme montré dans (8a-8b), et en suivant la dérivation classique de l'algorithme FRLS, les

équations de mise à jours du gain de kalman dual extrapolé $\tilde{\mathbf{c}}_{L+1,t}$ et la variable $\gamma_{L,t}$ sont trouvées et données par:

$$\tilde{\mathbf{c}}_{L+1,t} = \begin{bmatrix} 0 \\ \tilde{\mathbf{c}}_{L,t-1} \end{bmatrix} - \begin{bmatrix} \tilde{\mathbf{s}}_{N+1,t} \\ \mathbf{0}_{L-N} \end{bmatrix} \quad (\text{II.24})$$

$$\begin{bmatrix} \tilde{\mathbf{c}}_{L,t} \\ 0 \end{bmatrix} = \tilde{\mathbf{c}}_{L+1,t} - \begin{bmatrix} \mathbf{0}_{L-N} \\ \tilde{\mathbf{u}}_{N+1,t} \end{bmatrix} \quad (\text{II.25})$$

$$\delta_{L,t} = \delta_{L,t-1} + \tilde{s}_{N+1,t}^1 \bar{e}_{N,t} + \tilde{u}_{N+1,t}^{N+1} \bar{r}_{N,tD} \quad (\text{II.26})$$

avec:

$$\tilde{\mathbf{s}}_{N+1,t} = \frac{\bar{e}_{N,t}}{\lambda \alpha_{N,t-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{N,t-1} \end{bmatrix} \quad (\text{II.27})$$

$$\tilde{\mathbf{u}}_{N+1,t} = \frac{\bar{r}_{N,t-1}}{\lambda \alpha_{N,t-1}} \begin{bmatrix} -\mathbf{b}_{N,t-1} \\ 1 \end{bmatrix} \quad (\text{II.28})$$

Ou $\bar{e}_{N,t}$ et $\bar{r}_{N,t}$ sont les erreurs de prédictions aller et retour qui sont calculées par convolution (plus de détails sera donné sur la variance de l'erreur de prédiction retour dans la section4).

Toutes les quantités d'ordre N et $N+1$ sont calculées par la résolution du problème de prédiction d'ordre N .

Les deux vecteurs $\tilde{\mathbf{s}}_{N+1,t}$ et $\tilde{\mathbf{u}}_{N+1,t}$ sont disponibles dans la section de prédiction qui est incluse dans l'algorithme final. $\tilde{s}_{N+1,t}^1$ est la première valeur dans le vecteur $\tilde{\mathbf{s}}_{N+1,t}$, et $\tilde{u}_{N+1,t}^{N+1}$ est la $N+1^{\text{th}}$ valeur du vecteur $\tilde{\mathbf{u}}_{N+1,t}$.

Finalement, avec l'algorithme *FNTF*, la variable de vraisemblance $\gamma_{L,t}$ et l'erreur aller *a priori* $\bar{e}_{N,t}$ sont obtenues par les équations suivantes:

$$\bar{e}_{N,t} = x_t - \mathbf{a}_{N,t-1}^T \mathbf{x}_{N,t-1} \quad (\text{II.29})$$

$$\gamma_{L,t} = \frac{1}{1 - \delta_{L,t}} \quad (\text{II.30})$$

Ce paragraphe nous a permis d'avoir un algorithme appelé FTF12N2L [29] qui comporte deux parties de prédictions d'ordre N extraites de l'algorithme FTF, l'une utilisant les données disponibles à l'instant t et l'autre les données retardées disponibles à l'instant t_D .

La liste des équations correspondantes à l'algorithme FNTF12N2L stabilisé est donnée par la table II.1. Nous avons modifié l'algorithme original de [29] en lui appliquant la procédure de stabilisation d'écrite dans [36]. Cette dernière est basée sur la rétroaction des erreurs numériques dans le calcul des variables de prédiction rétrogrades.

II.5. Stabilisation numérique de l'algorithme FNTF

Pour stabiliser l'algorithme de filtrage adaptatif FNTF, nous avons utilisé une technique de stabilisation numérique qui a été proposée dans [35], cette dernière est basée sur la rétroaction des erreurs numériques dans le calcul de l'erreur de prédiction retour. Cette méthode est basée sur les équations suivantes:

$$\bar{r}_{N,t}^p = -\lambda^{-N+1} \alpha_{N,t} \tilde{c}_{N+1,t}^{N+1} \quad (\text{II.31})$$

Ceci permet d'éviter le calcul de la variance retour $\beta_{N,t}$, qui est numériquement instable. Cependant, ceci ne stabilise pas les cinq algorithmes. Pour les stabiliser, la méthode décrite dans [36] a été adaptée et appliquée, c'est-à-dire, deux *BPEs* ont été employées pour calculer la variable de vraisemblance et predicteur retour [35]:

$$\begin{aligned} \bar{r}_{N,t}^b &= \bar{r}_{N,t}^c + \mu^b \xi_{N,t} \\ \bar{r}_{N,t}^\gamma &= \bar{r}_{N,t}^c + \mu^\gamma \xi_{N,t} \end{aligned} \quad (\text{II.32})$$

$$\quad \quad \quad (\text{II.33})$$

Ou la variable de contrôle $\xi_{N,t}$ est donnée par:

$$\xi_{N,t} = \bar{r}_{N,t}^c - \bar{r}_{N,t}^p = \bar{r}_{N,t}^c + \lambda^{-N+1} \alpha_{N,t} \tilde{c}_{N+1,t}^{N+1} \quad (\text{II.34})$$

et $\bar{r}_{N,t}^c$ est la *BPE* calculée par convolution, c'est-à-dire,

$$\bar{r}_{N,t}^c = x_{t-N} - \mathbf{b}_{N,t-1}^T \mathbf{x}_{N,t} \quad (\text{II.35})$$

sont utilisés, respectivement, pour actualiser les variables de la partie de $\bar{r}_{N,t}^\gamma$ et $\bar{r}_{N,t}^b$. On note que prédiction aller et retour de chaque algorithme.

permettent la modification et le contrôle de la propagation des erreurs μ^b et μ^γ . Les paramètres numériques dans l'algorithme. On a choisie

$$\mu^\gamma = \mu^b = 1$$

Avec ce choix, il peut être montré que le predicteur retour est numériquement stable sous la condition suivante [35,36,37]:

$$1 > \lambda > \frac{4N+5}{4N+7} \quad (\text{II.36})$$

ces équations qui permettent la stabilité de l'algorithme sont repérées dans la table II.1 par (S1,S2,S3).

Cet algorithme (FTF12N2L), nécessite $12N+2L$ multiplications (en négligeant les opérations sur les scalaires), soit un gain théorique en complexité de l'ordre d'un facteur 4 sur le FTF lorsque $N \ll L$, la complexité théorique est donc approximativement celle du NLMS.

• **Partie de prédiction de l’algorithme FTF12N2L à l’instant t.**

Les équations repérées (S1,S2,S3) correspondent à la technique de stabilisation numérique [36,37]

On pose $X_{N,t} = [x_t, x_{t-1} \dots x_{t-N+1}]^T$; $\tilde{C}_N, \tilde{C}_{N+1}$: intermédiaire de calcul.

$\bar{e}_{N,t} = x_t - a_{N,t-1}^T X_{N,t-1}$ $e_{N,t} = \gamma_{N,t-1} \bar{e}_{N,t}$ $\alpha_{N,t} = \lambda \alpha_{N,t-1} + \bar{e}_{N,t} e_{N,t}$ $\gamma_{N+1,t} = \frac{\lambda \alpha_{N,t-1}}{\alpha_{N,t}} \gamma_{N,t-1}$ $\tilde{S}_{N+1,t} = \frac{\bar{e}_{N,t}}{\lambda \alpha_{N,t-1}} \begin{bmatrix} 1 \\ -a_{N,t-1} \end{bmatrix}$ $\tilde{C}_{N+1,t} = \begin{bmatrix} 0 \\ \tilde{C}_{N,t-1} \end{bmatrix} - \tilde{S}_{N+1,t}$ $a_{N,t} = a_{N,t-1} - e_{N,t} \tilde{C}_{N,t-1}$ $\bar{r}_{N,t}^{conv} = x_{t-N} - b_{N,t-1}^T X_{N,t} \quad (S1)$	$\xi_{N,t} = \bar{r}_{N,t}^{conv} + \lambda \beta_{N,t-1} \tilde{C}_{N+1,t}^{N+1} \quad (S2)$ $\bar{r}_{N,t} = \bar{r}_{N,t}^{conv} + \xi_{N,t} \quad (S3)$ $\gamma_{N,t} = \frac{\gamma_{N+1,t}}{1 + \gamma_{N+1,t} \bar{r}_{N,t} \tilde{C}_{N+1,t}^{N+1}}$ $r_{N,t} = \gamma_{N,t} \bar{r}_{N,t}$ $\tilde{U}_{N+1,t} = \tilde{C}_{N+1,t}^{N+1} \begin{bmatrix} -b_{N,t-1} \\ 1 \end{bmatrix}$ $\begin{bmatrix} \tilde{C}_{N,t} \\ 0 \end{bmatrix} = \tilde{C}_{N+1,t} - \tilde{U}_{N+1,t}$ $b_{N,t} = b_{N,t-1} - r_{N,t} \tilde{C}_{N,t}$ $\beta_{N,t} = \lambda \beta_{N,t-1} + \bar{r}_{N,t} r_{N,t}$
--	--

Table II.1 : Partie de prédiction de l’algorithme FTF12N2L à l’instant t.

II.6. Initialisation de l’algorithme FNTF

L’initialisation de l’algorithme FNTF suit la procédure optimale adaptée au préfenêtrage. Les $\tilde{C}_{L,0}, \tilde{C}_{N,-L+N}, \tilde{C}_{N,0}$ et le gain de KALMAN dual $b_{N,-L+N}, a_{N,-L+N}, b_{N,0}, a_{N,0}$ predicteurs sont mis à un. $\gamma_{L,t}, \gamma_{N,-L+N}, \gamma_{N,0}$ mis à zéro, tandis que les variables de vraisemblances La table II.2 résume l’initialisation des variances des erreurs de prédictions.

$L,0 = \lambda^L e_0$	$L,0 = \lambda^{L-N} e_0$	$L,-L+N = \lambda^N e_0$	$N,-L+N = e_0$
-----------------------	---------------------------	--------------------------	----------------

Table II.2 : Initialisation des variances des erreurs de prédiction

Où e_0 est une estimation a priori de la variance σ_x^2 de l’entrée.

II.7. L'algorithme Fast Newton Simplifié (FNS)

L'algorithme de 'fast Newton simplifié' (FNS) [39] est le résultat de la combinaison de l'algorithme du gradient stochastique LMS avec l'algorithme de 'Fast Newton Transversal filters' FNTF12N2L

Cette combinaison a pour but d'exploiter les performances des deux types d'algorithmes, afin de donner un algorithme qui possède une complexité de calcul comparable à celle du gradient stochastique est une vitesse de convergence proche de celle de l'algorithme FTF12N2L.

Ceci est réalisé en utilisant la structure de l'algorithme LMS, dans la partie de prédiction de l'algorithme FTF12N2L, ce qui donne naissance à un nouveau algorithme appelé 'Fast Newton Simplifié FNS.

Les erreurs de prédictions directe et retour sont données par les équations suivantes :

$$e_{M,t}^f = x_t - A_{M,t-1}^T X_{M,t-1} \quad (\text{II.37})$$

$$e_{M,t}^b = x_{t-M} - B_{M,t-1}^T X_{M,t} \quad (\text{II.38})$$

Où $A_{M,t-1}$ et $B_{M,t-1}$ sont les deux prédicteurs aller et retour, et sont donnés par :

$$A_{M,t} = A_{M,t-1} + \mu^f e_{M,t}^f X_{M,t-1} \quad (\text{II.39})$$

$$B_{M,t} = B_{M,t-1} + \mu^b e_{M,t}^b X_{M,t} \quad (\text{II.40})$$

Les deux nouveaux paramètres μ^f et μ^b qui sont les pas d'adaptation des prédicteurs aller et retour sont sélectionnés par simulation.

La partie de filtrage et de la mise à jour des coefficients du filtre transverse adaptatif est donnée par les équations suivantes.

$$\varepsilon_{N,t} = d_t - H_{N,t-1}^T X_{N,t} \quad (\text{II.41})$$

$$H_{N,t} = H_{N,t-1} + \mu \varepsilon_{N,t} C_{N,t} \quad (\text{II.42})$$

avec :

$\varepsilon_{N,t}$: est l'erreur de filtrage adaptatif

d_t : est le signal désiré

$C_{N,t}$: est le gain de Kalman dual

μ : est le pas d'adaptation de la partie de filtrage

Le pas d'adaptation de la partie de filtrage est donné par :

$$\mu = \frac{2\sigma_{mis}^2}{\sigma_x^2 N + \alpha_{M,t-N+M}^d} \quad (\text{II.43})$$

où :

σ_x^2 : est la variance du signal d'entrée $x(n)$.

α_M^d : est la puissance de l'erreur de prédiction directe/indirecte d'ordre M des prédicteurs aller/retour D_{M+1} .

σ_{mis}^2 : est le disajustement des paramètres qui est donné par :

$$\sigma_{mis}^2 = \frac{\sigma_{ex}^2(\infty)}{\sigma_{min}^2} \tag{II.44}$$

Où :

$\sigma_{ex}^2(\infty)$: est la variance de l'erreur de filtrage supplémentaire.

σ_{min}^2 : est la variance de l'erreur de filtrage minimale (erreur quadratique moyenne minimale).

Les nouveaux vecteurs de la prédiction aller/retour d'ordre M+1 sont donnés comme suit :

$$D_{M+1,t-1} = \begin{bmatrix} I \\ -A_{M,t-1} \end{bmatrix} \tag{II.45}$$

$$D_{M+1,t-1} = \begin{bmatrix} -B_{M,t-1} \\ I \end{bmatrix} \tag{II.46}$$

Les équations qui vont permettre de calculer le gain de Kalman sont données sous formes deux versions différentes, la première version utilise les deux prédicteurs aller et retour pour la mise à jour du gain de Kalman, cette dernière est donnée par la relation suivante.

Version 1 :

$$\begin{bmatrix} C_{N,t} \\ * \end{bmatrix} = \begin{bmatrix} x_t \\ C_{N,t-1} \end{bmatrix} + Ke_{M,t}^d \begin{bmatrix} D_{M+1,t-1} \\ 0_{N-M} \end{bmatrix} - Ke_{M,n_M}^d \begin{bmatrix} 0_{N-M} \\ D_{M+1,n_M-1} \end{bmatrix} \tag{II.47}$$

Où

$$e_{M,t}^d = e_{M,t}^f$$

$$e_{M,n_M}^d = e_{M,t}^b$$

$$\text{et } n_M = t - N + M$$

Une autre version de cet algorithme qui permet encore la réduction de la complexité de calcul est la version deux. Cette dernière consiste à éliminer toute la partie de prédiction retour, en utilisant seulement le prédicteur aller et l'erreur de prédiction aller dans la mise à jour du gain de Kalman.

Cette version est illustrée par la relation suivante :

Version 2 :

$$\begin{bmatrix} C_{N,t} \\ * \end{bmatrix} = \begin{bmatrix} x_t \\ C_{N,t-1} \end{bmatrix} + Ke_{M,t}^d \begin{bmatrix} D_{M+1,t-1} \\ 0_{N-M} \end{bmatrix} \tag{II.48}$$

Les équations de fonctionnement de l’algorithme FNS sont listées dans la table II.3 suivant :

<p><i>Variable à l’instant n-1: $W_N(n-1), C_N(n-1), X_N(n-1)$</i></p> <p>$A_M(n-1), B_M(n-1), A_M(n_M-1), B_M(n_M-1)$, $(n_M = n - N + M)$</p>	
<p><i>Nouvelle information: $d(n), x(n)$</i></p>	
<p><i>Adaptation du prédicteur directe et indirecte en utilisant LMS :</i></p>	
$e_M^f(n) = x(n) - A_M^T(n-1)X_M(n-1)$ $A_M(n) = A_M(n-1) + \mu^f e_M^f(n)X_M(n-1)$ $D_{M+1}(n-1) = \begin{bmatrix} I \\ -A_M(n-1) \end{bmatrix}$ $e_M^d(n) = e_M^f(n)$	$e_M^b(n) = x(n-M) - B_M^T(n-1)X_M(n)$ $B_M(n) = B_M(n-1) + \mu^b e_M^b(n)X_M(n)$ $D_{M+1}(n-1) = \begin{bmatrix} -B_M(n-1) \\ I \end{bmatrix}$ $e_M^d(n) = e_M^b(n)$
<p><i>calcul du gain C_L :</i></p>	
<p><i>version 1 :</i></p> $\begin{bmatrix} C_L(n) \\ * \end{bmatrix} = \begin{bmatrix} x(n) \\ C_L(n-1) \end{bmatrix} + Ke_M^d(n) \begin{bmatrix} D_{M+1}(n-1) \\ 0_{L-M} \end{bmatrix} - Ke_M^d(n_M) \begin{bmatrix} 0_{L-M} \\ D_{M+1}(n_M-1) \end{bmatrix}$	
<p><i>version 2 :</i></p> $\begin{bmatrix} C_L(n) \\ * \end{bmatrix} = \begin{bmatrix} x(n) \\ C_L(n-1) \end{bmatrix} + Ke_M^d(n) \begin{bmatrix} D_{M+1}(n-1) \\ 0_{L-M} \end{bmatrix}$	
<p><i>Partie de filtrage :</i></p> $\varepsilon_L(n) = d(n) - H_L^T(n-1)X_L(n)$ $H_L(n) = H_L(n-1) + \mu \varepsilon_L(n)C_L(n)$	

Table II.3 : listing de l’algorithme ‘Fast Newton ‘ simplifié.

II.8. Complexité des calculs

Le tableau II.1 suivant donne la complexité de calcul (nombre d'opération numérique : les multiplications et les divisions) des algorithmes d'identification adaptative suivants:

- Algorithme des moindres carrés rapide FTF,
- Algorithme de "Fast Newton" FNTF12N2L,
- Algorithme de "Fast Newton Simplifié" FNS,
- Algorithme du gradient stochastique normalisé NLMS .

Ces complexités des calculs sont données en fonction de N et de L où L représente la taille du filtre transversal et N représente la taille de la partie de prédiction.

(dans cette table, une division est prise égale à 16 multiplications).

Type d'algorithmes	Nombre de Multiplications	Nombre de Divisions	Complexité (MADPR)
FTF8L	8L	3	8L+48
FTF12N2L	2L+12N	7	2L+12N+112
FNS(version :1)	2L+6N	0	2L+6N
FNS(version :2)	2L+3N	0	2L+3N
NLMS	2L	1	2L+16

Table II.4 : complexité de calcul des algorithmes:

FTF, FTF12N2L, FNS et NLMS.

Le **tableau II.2** donne un exemple typique de la complexité des calculs des algorithmes d'identification adaptatif cités au dessus par rapport à celle du NLMS pour une taille du filtre transverse égale à ($L=256$) et un ordre de prédiction égale à ($N=16$).

Types d'algorithmes	Nombre des Multiplications	Par rapport à /NLMS
FTF8L	2096	4
FTF12N2L	816	1.54
FNS(version :1)	608	1.15
FNS(version :2)	560	1.06
NLMS	528	1

Table II.5 : la complexité pour $L= 256$, $N= 16$

II.9. Conclusion

Nous avons présenté des algorithmes de filtrage adaptatif transverse de types de moindres carrés rapides récemment développés avec une complexité des calculs réduite pour l'identification des réponses impulsionnelles longues. Cette présentation de l'algorithme FNTF est basée sur une approche qui ne fait référence qu'à la prédiction linéaire ce qui nous semble plus abordable que la présentation originale.

Une forme simplifiée de l'algorithme de Newton rapide FNS a été présentée dans le but de réduire encore la complexité des calculs, cette simplification est présentée sous formes de deux versions.

Les nouvelles versions algorithmiques du FNTF, FNS et FTF et que nous avons présentées et dont cin versions ont été publiées dans le journal de Signal Processing, sont basées sur les versions de base que nous les avons détaillées dans ce chapitre. Ces nouveaux algorithmes ainsi que leurs études théoriques vont être présentées et raffinées dans le chapitre III.

Chapitre III

Nouvelles versions des algorithmes des moindres carrés transversaux rapides type Newton

III.1. Introduction

L'identification adaptative des systèmes linéaires paramétrés en termes de leurs réponses impulsionnelles a été étudiée intensivement et pour longtemps, et beaucoup d'algorithmes récursifs ont été proposés pour résoudre ce problème [40,41].

Les algorithmes les plus connus pour le filtrage adaptatif transversal sont les moindres carrés moyennes (*LMS* ou *NLMS*) et les algorithmes des moindres carrés récursifs (*RLS*).

Dans l'algorithme *LMS* ou sa version normalisée *NLMS*, les propriétés de convergence sont affectées par les statistiques du signal d'entrée, alors que dans les algorithmes de type *RLS*, on résout le problème des moindres carrés (*LS*) récursivement et convergent rapidement même avec les entrées colorées, puisqu'ils estiment l'inverse de la matrice de covariance du signal. Les algorithmes *RLS* rapides (*FRLS*) ont été proposés afin de réduire la complexité de calcul lourde $O(L^2)$ des algorithmes *RLS* et la rendre une fonction linéaire avec l'ordre du filtre L [11].

Afin de couvrir le gap entre les deux classes algorithmiques extrêmes stochastique-Newton, plusieurs algorithmes ont été proposés dans la littérature tel que l'algorithme FTF (Fast transversal Filter), l'algorithme de Newton rapide (FNNTF) [29], l'algorithme de l'ordre indéterminé du RLS (URLS [48], [49] et l'algorithme de projection affine [50] et sa version rapide [51].

Dans cette thèse nous nous intéresserons aux algorithmes de type moindres carrés rapides de type Newton FNNTF qui sont numériquement stables.

La particularité de cette famille d'algorithme est qu'elle exploite une supposition que les prédicteurs linéaires comme ceux utilisés dans les algorithmes rapides de type RLS peuvent être limités à la longueur N (nouvelle taille des prédicteurs aller et retour) sans pertes en performance.

L'algorithme FNNTF utilise cette propriété pour réduire la complexité de calcul de $7L$ à $2L+7N$. Cette simplification nécessite un accroissement de la mémoire de $N(L-N)$ mémoires supplémentaires. Alternativement, l'utilisation de mémoires additionnelles peut être évitée avec une augmentation de complexité à $2L+12N$. Quand $N=1$ le FNNTF, se simplifie à l'algorithme du NLMS et quand $N=L$, le FNNTF devient le FRLS

Dans cette contribution, on propose cinq nouvelles versions de l'algorithme de filtrage adaptatif transversal de Newton rapide (FNNTF), deux nouvelles versions de l'algorithme FTF et une nouvelle version par bloc de l'algorithme FNS (FNNTF simplifiée). Chaque nouvel algorithme est présenté et comparé avec la version originale.

Les résultats d'une étude comparative de la convergence et de la capacité de poursuite entre les cinq versions proposées de l'algorithme FNNTF et leurs versions originales sont également présentés. L'algorithme FNS par bloc et les deux versions de l'algorithme FTF seront détaillés en présentant leurs résultats de simulations. Le reste de ce chapitre est organisé comme suite:

Les algorithmes proposés de l'algorithme FNNTF sont donnés dans la section III.2, la stabilité et les issues de complexité sont discutées dans les paragraphes III.3 et III.4, les initialisations de chaque algorithme sont données dans la section III.5. La vitesse de convergence et l'analyse de la capacité de poursuite des algorithmes proposés sont discutées dans la section III.6, des résultats de simulations sont présentés dans la section III.7. L'algorithme FNS par bloc est présenté dans la section III.8, une description des deux nouvelles versions de l'algorithme FTF est donnée dans la section III.9 et finalement une conclusion est donnée dans la section III.10.

III.2. Les nouvelles versions de l'algorithme FNNTF

dans cette section, nous allons présenter cinq nouvelles versions de l'algorithme de filtrage adaptatif de type de Newton rapide (FNNTF). Une brève présentation préliminaire de ces algorithmes a été donnée dans la conférence d'ICASSP 2004 [40] et une autre version plus détaillée est publiée dans le journal Signal Processing [54].

Le premier Algorithme est basé sur une simple modification de la partie de filtrage par l'introduction d'un paramètre accélérateur. Le deuxième Algorithme est basé sur l'utilisation d'une technique de découpage temporelle dans l'actualisation des coefficients du filtre. Le troisième Algorithme est une modification du deuxième Algorithme, par l'utilisation de l'erreur de filtrage finale pour actualiser les coefficients du filtre. Le quatrième et cinquième Algorithmes sont basés, respectivement, sur la combinaison des propriétés du premier Algorithme avec les propriétés du deuxième et le troisième Algorithmes. Ces nouveaux Algorithmes sont proposés pour l'amélioration de vitesse de convergence de la version originale de l'Algorithme FNNTF pour l'identification des réponses impulsionnelles acoustiques, et d'améliorer aussi la capacité de poursuite lorsque le system varie dans le temps. Une étude comparative de chaque Algorithme avec la version originale de l'Algorithme FNNTF est présentée.

III.2.1. Algorithme 1

Afin d'améliorer les performances de l'algorithme FNNTF présenté dans la section 2 dans le cas des canaux de couplages acoustiques et des entrées non stationnaires, nous avons introduits des modifications simples qui n'altèrent pas les bonnes propriétés numériques de l'algorithme. La capacité de poursuite est améliorée par le contrôle du gain d'adaptation dans la partie de filtrage [43], [44]. Pour ce faire, un facteur de pondération variable dans le temps μ_t est introduit dans le calcul de la partie de filtrage:

$$\mathbf{h}_{L,t} = \mathbf{h}_{L,t-1} - \mu_t \bar{\mathbf{e}}_{L,t} \mathbf{c}_{L,t} \quad (\text{III.1})$$

Avec $\mathbf{c}_{L,t} = \gamma_{L,t} \tilde{\mathbf{c}}_{L,t}$ et $\gamma_{L,t}$ sont fournis par la partie d'extrapolation de l'algorithme *FNTF*. Le paramètre μ_t peut être avantageusement rendu dépendant du temps. Ceci est clairement vu si l'effet de μ_t est examiné sur l'erreur de filtrage *a posteriori* après l'itération à l'instant t [43]:

$$\varepsilon_{L,t} = y_t - \mathbf{h}_{L,t}^T \mathbf{x}_{L,t} = (1 + \mu_t \mathbf{c}_{L,t}^T \mathbf{x}_{L,t}) \bar{\varepsilon}_{L,t} \quad (\text{III.2})$$

A partir de cette équation, il est clair que le choix suivant du paramètre μ_t annule exactement l'erreur de filtrage *a posteriori* :

$$\mu_t = -1 / \mathbf{c}_{L,t}^T \mathbf{x}_{L,t} = 1 / (1 - \gamma_{L,t}) \quad (\text{III.3})$$

Dans le but de contrôler l'effet de ce facteur de pondération, un paramètre constant μ est introduit dans (III.3):

$$\mu_t = 1 / (1 - \mu \gamma_{L,t}) \quad (\text{III.4})$$

Le paramètre μ permet le contrôle de la performance du nouveau algorithme. La variable de vraisemblance $\gamma_{L,t}$ prend ces valeurs entre 0 et 1 [11]; cependant, le paramètre μ doit être inférieure ou égale à 1 pour assurer que μ_t garde une valeur positive. On note que le choix $\mu = 0$ conduit à la version originale de l'algorithme *FNTF*. Finalement, avec une seule division supplémentaire, le nouvel algorithme *FNTF* (Algorithme 1) est donné par:

$$\mathbf{h}_{L,t} = \mathbf{h}_{L,t-1} - \frac{1}{1 - \mu \gamma_{L,t}} \mathbf{c}_{L,t} \bar{\varepsilon}_{L,t} \quad (\text{III.5})$$

On note que la condition de convergence et de stabilité de cet algorithme (Algorithme 1) est: $0 < \mu < 1$ et $\lambda > k$, ou k une constante qui va être décrite dans le paragraphe de la stabilisation numérique (section III.3).

III.2.2. Algorithme 2

Cet algorithme est basé sur une modification qui produit une structure de prédiction permettant à l'algorithme *FNTF* de fonctionner avec un facteur d'oubli exponentiel λ inférieure que celui de la structure de base initiale.

La technique suggérée (Algorithme 2) permet de diminuer implicitement le facteur d'oubli exponentiel des deux parties de prédiction aller et retour de l'algorithme *FNTF*, car, l'algorithme *FNTF* contient les deux parties de prédictions de l'algorithme du filtre transversal rapide (*FTF*, Fast Transversal Filter) et la même partie de filtrage comme dans le *FTF* (voir section III.2).

La partie d'extrapolation de l'algorithme *FNTF* consiste essentiellement en décalages de vecteurs plus une division que nous comptons dans la partie de filtrage.

Le vecteur du gain d'adaptation dans l'algorithme *FNTF* est donné près $\mathbf{g}_{L,t} = \mathbf{c}_{L,t} = \gamma_{L,t} \tilde{\mathbf{c}}_{L,t}$, ou les vecteurs $\mathbf{c}_{L,t}$ (gain de Kalman), $\tilde{\mathbf{c}}_{L,t}$ (gain de Kalman dual) et $\gamma_{L,t}$ (la variable de

vraisemblance) peuvent être calculés dans la partie de prédiction (aller et retour) et dans les parties d'extrapolations de l'algorithme FNTF. Cet algorithme nécessite le calcul du même gain de Kalman dual, ceci correspond au cas de pré-fenêtrage, c'est-à-dire $x_t = y_t = 0$, pour $t \leq 0$ avec un facteur d'oubli exponentiel $\lambda < 1$.

Les définitions du gain de Kalman dual et de la variable de vraisemblance sont respectivement données par (II.4) et (II.5). On a montré dans [44] que les propriétés de la capacité de poursuite de l'algorithme FTF n'est pas meilleure en raison de la contrainte sur le choix du facteur d'oubli exponentiel λ qui doit satisfaire la condition suivante:

$$0 < \mu < 1 \text{ et } \lambda > k ,$$

ou k est une constante qui va être décrites dans la section III.5.

Afin de surmonter cette contrainte sur les valeurs du facteur d'oubli exponentiel λ de l'algorithme FNTF (l'algorithme FNTF est construit par deux parties de prédiction de l'algorithme FTF), on emploie une nouvelle structure qui remplace la structure de l'algorithme global par P Algorithmes de longueurs égales ou inégales, fonctionnant chacun sur un nombre réduit de paramètres.

Cette technique mène à réaliser P algorithmes FNTF fonctionnant en série comme montré par la structure représentée dans la figure III.1, pour le cas de $\mu=0$. Les P algorithmes adaptatifs sont adaptés par les erreurs de filtrage locales pour filtrer l'écho y_t . Les équations de mise à jour des P filtres transversaux sont données par:

$$\mathbf{h}_{L_i,t} = \mathbf{h}_{L_i,t-1} - \bar{\mathbf{e}}_{i,t} \mathbf{c}_{L_i,t} , \quad i = 1, 2, \dots, P \quad (\text{III.6})$$

Ou:

$$\sum_{i=1}^P L_i = L \quad (\text{III.7})$$

et P représente le nombre de sous-divisions. Ce paramètre est choisi par simulation et usuellement pris entre 2 et 4 (si $P = 1$, l'algorithme 2 retrouve la version classique de l'algorithme FNTF). La longueur des filtres peut être égale à L/P (L représente la longueur totale du filtre, elle est donnée par (III.7)). Lorsque le facteur L/P est entier, les longueurs des filtres sont égales à une dimension de L_i et $L = P \times L_i$ ($i=1, 2, \dots, P$). et lorsque L/P n'est pas entier, les longueurs des filtres sont inégales et la longueur totale du filtre est donnée par (III.7).

On note que le gain de Kalman de la section i , $\mathbf{c}_{L_i,t}$ avec dimension L_i , est calculé par un algorithme FNTF avec un facteur d'oubli exponentiel λ_i basé sur L_i et sur le signal d'entrée x_t retardé par :

$$D_i = L_1 + L_2 + \dots + L_{i-1} \text{ échantillons.}$$

Cette nouvelle structure a été appliquée à l'algorithme FNTF, et elle a produit l'algorithme 2. l'erreur de filtrage de la section i est calculée comme suite:

$$\bar{\varepsilon}_{i,t} = \bar{\varepsilon}_{i-1,t} - \mathbf{h}_{L_i,t-1}^T \mathbf{x}_{L_i,t}, \quad (\text{III.8})$$

avec $\bar{\varepsilon}_{0,t} = y_t$ et,

$$\mathbf{x}_{L_i,t}^T = (x_{t-D_i} \dots x_{t-D_i-L_i+1}), \quad (\text{III.9})$$

avec $L_0 = 0$.

La structure de cet algorithme est donnée par la figure III.1 avec $\mu=0$.

III.2.3. Algorithme 3

Cet algorithme est une modification de l'algorithme FNTF modifié précédemment dans la section III.2.2 (algorithme 2). L'algorithme 3 est basé sur l'utilisation de l'erreur de filtrage totale(ou finale) pour la mise à jour des P filtres adaptatifs .

Quand nous examinons la structure de l'algorithme 2, nous concluons que les erreurs de filtrage que nous employons pour actualiser les coefficients locaux du filtre de chaque subdivision ont des valeurs décroissantes, i.e.

$$\bar{\varepsilon}_1 > \bar{\varepsilon}_2 > \dots > \bar{\varepsilon}_{p-1} > \bar{\varepsilon}_p.$$

Nous notons également que la partie non identifiée de y_t dans cette structure introduit un biais dans le calcul des coefficients de la première subdivision, alors que le vecteurs du signal x_{L_i} de chaque subdivision n'a aucun élément en commun.

L'idée fondamentale de l'algorithme 3 est d'employer les plus petites erreurs de filtrage pour actualiser tous les coefficients du filtre de chaque subdivision. Cette technique, une fois appliquée à la structure 1 de l'algorithme 2, permet d'améliorer l'adaptation des coefficients des filtres des premières subdivisions davantage que les dernières subdivisions de la structure, mais elle améliore (elle sera montrée par la simulation) le comportement global de l'algorithme FNTF (vitesse de convergence et capacités de poursuite).

Les équations de base de cette structure sont obtenues à partir de l'algorithme 2 et en substituant les erreurs locales de (III.6) par l'erreur de filtrage finale obtenu à la sortie de la dernière subdivision:

$$\bar{\varepsilon}_{L_i,t} = y_t - \sum_{i=1}^P \mathbf{h}_{L_i,t-1}^T \mathbf{x}_{L_i,t} \quad (\text{III.10})$$

ou:

$\bar{\varepsilon}_{L_i,t}$ représente l'erreur de filtrage finale ;

et h_{L_i} ($i=1,2,3\dots P$) sont les coefficients locales du filtre de chaque subdivisions.

En conclusion, la version classique de l'algorithme FNTF est obtenue à partir de l'algorithme 3 en mettant le nombre subdivision $P=1$ pour être la même qu'avec l'algorithme 2. Nous notons également que la longueur de filtre de chaque subdivision de cet algorithme est calculée comme dans l'algorithme 2, et le paramètre P est pris entre 2 et 4 et choisi par la simulation.

La structure de cet algorithme est montré par la figure III.2 pour le cas $\mu=0$.

III.2.4. Algorithme 4

Cet algorithme est inspirée de la combinaison entre les caractéristiques de l'algorithme 1 et celui de l'algorithme 2. donc, cet algorithme regroupe les avantages des deux techniques, donc il permet une amélioration des performances des deux algorithmes 1 et 2.

Donc, comme nous l'avons expliqué précédemment, les équations de l'algorithme 4 sont basées sur l'équation de mise à jour de l'algorithme 1 et celle de la structure fendue de l'algorithme 2, c-à-d. la même structure que dans Fig.III.1, avec les équations (III.8)-(III.9) [40]:

Ainsi un scalaire variable μ est introduit dans les parties de filtrages.

$$h_{L_i,t} = h_{L_i,t-1} - \frac{1}{1 - \mu \gamma_{L_i,t}} c_{L_i,t} \bar{\epsilon}_{i,t} \quad i = 1, 2, \dots, P \tag{III.11}$$

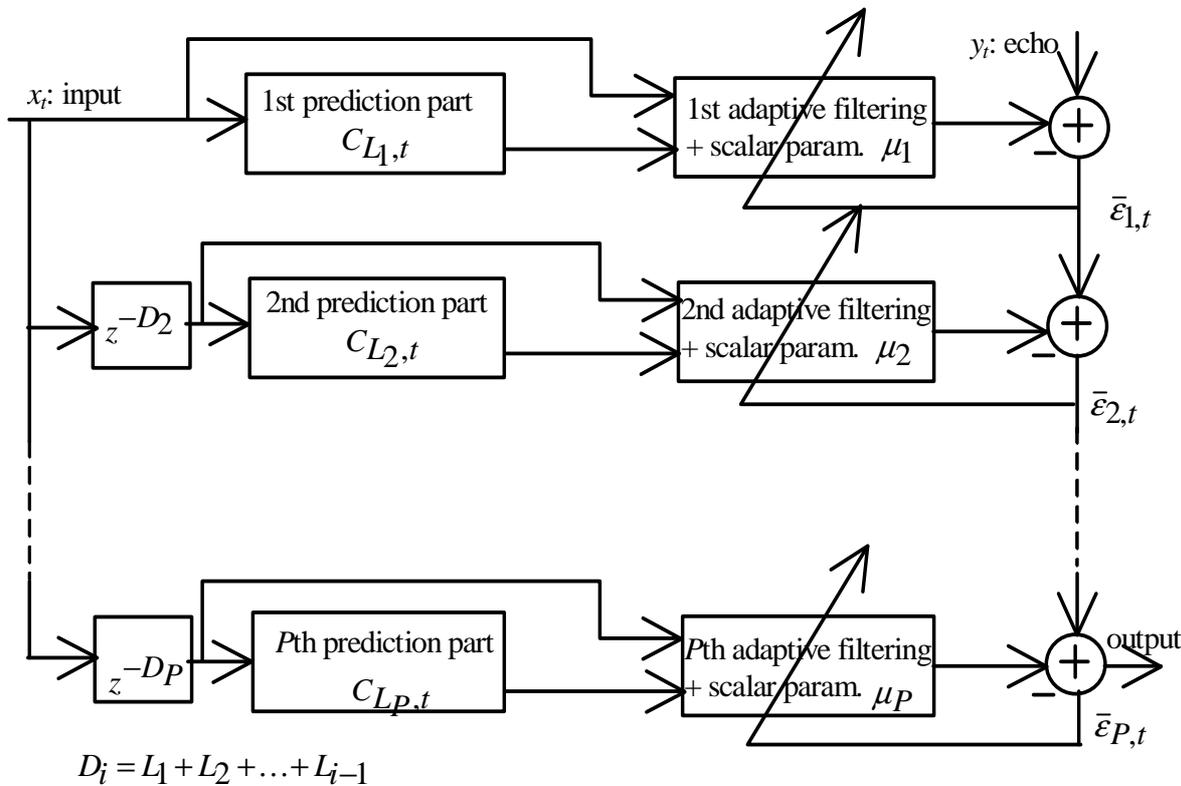


Figure III.1. Division temporelle de l'algorithme de FNTF, première approche
 Pour $\mu=0$: Algorithme 2, pour le $\mu = \mu_i$: Algorithme 4.

III.2.5. Algorithme 5

Cet algorithme est basé sur une équation de mise à jour comme dans l'algorithme 1 et sur la structure de l'algorithme 3, c'est-à-dire comme dans Fig.III.2, avec l'équation (III.10). Ainsi une variable scalaire μ est introduite dans les parties de filtrages.

$$h_{L_i,t} = h_{L_i,t-1} - \frac{1}{1 - \mu \gamma_{L_i,t}} c_{L_i,t} \bar{\epsilon}_{L,t}, \quad i = 1, 2, \dots, P \tag{III.12}$$

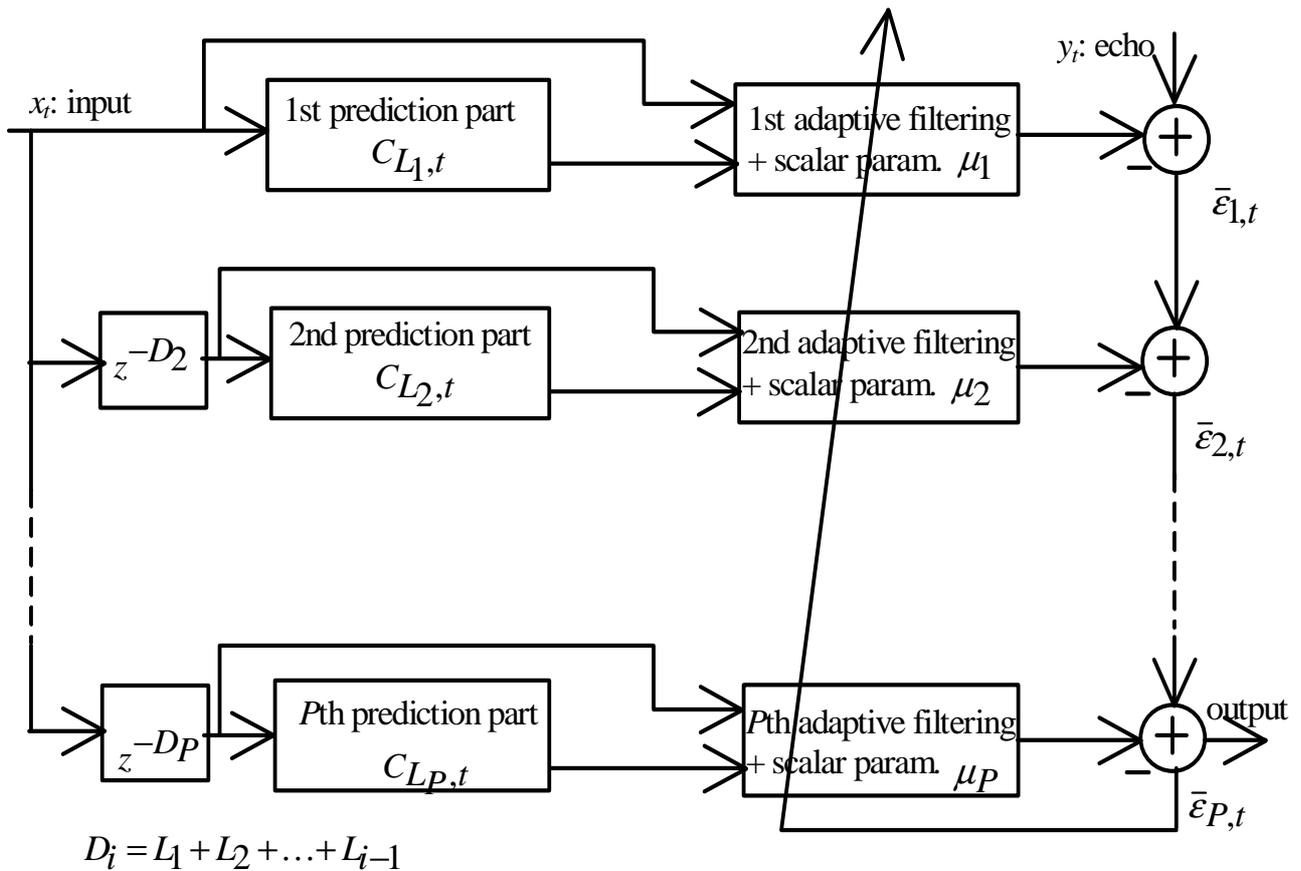


Figure III.2. Division temporelle de l'algorithme *FNTF*, Deuxième approche.
 Pour $\mu=0$: Algorithme 3, for $\mu = \mu_i$: Algorithme 5.

III.3. Stabilisation numérique des nouveaux algorithmes

Une méthode de stabilisation numérique de base a été employée pour stabiliser les cinq algorithmes proposés [36], [45]. La différence entre ces nouveaux algorithmes et ceux de [40] et [46] se situe dans le calcul de l'erreur de prédiction retour *a priori* (BPE) dans les deux parties de prédictions de l'algorithme FNTF. La variance de l'erreur de prédiction aller $\alpha_{N,t}$ utilisée dans le calcul de *BPE* est donnée par:

$$\bar{r}_{N,t}^p = -\lambda^{-N+1} \alpha_{N,t} \tilde{c}_{N+1,t}^{N+1} \quad (\text{III.13})$$

Ceci permet d'éviter le calcul de la variance retour $\beta_{N,t}$, qui est numériquement instable. Cependant, ceci ne stabilise pas les cinq algorithmes. Pour les stabiliser, la méthode décrite dans [36] a été adaptée et appliquée, c'est-à-dire, deux *BPEs* ont été employées pour calculer la variable de vraisemblance et le prédicteur retour [35]:

$$\bar{r}_{N,t}^b = \bar{r}_{N,t}^c + \mu^b \xi_{N,t} \quad (\text{III.14})$$

$$\bar{r}_{N,t}^\gamma = \bar{r}_{N,t}^c + \mu^\gamma \xi_{N,t} \quad (\text{III.15})$$

ou la variable de contrôle $\xi_{N,t}$ est donnée par:

$$\xi_{N,t} = \bar{r}_{N,t}^c - \bar{r}_{N,t}^p = \bar{r}_{N,t}^c + \lambda^{-N+1} \alpha_{N,t} \tilde{c}_{N+1,t}^{N+1} \quad (\text{III.16})$$

et $\bar{r}_{N,t}^c$ est la *BPE* calculée par convolution, c'est-à-dire,

$$\bar{r}_{N,t}^c = x_{t-N} - \mathbf{b}_{N,t-1}^T \mathbf{x}_{N,t} \quad (\text{III.17})$$

On note que $\bar{r}_{N,t}^b$ et $\bar{r}_{N,t}^\gamma$ sont utilisés, respectivement, pour actualiser les variables de la partie de prédiction aller et retour de chaque algorithme.

Les paramètres μ^γ et μ^b permettent la modification et le contrôle de la propagation des erreurs numériques dans chaque algorithme.

On a choisie $\mu^\gamma = \mu^b = 1$. Avec ce choix, il peut être montré que le prédicteur retour est numériquement stable sous la condition suivante [35,36]:

$$1 > \lambda > \frac{4N+5}{4N+7} \quad (\text{III.18})$$

III.4. Évaluation de la complexité de calcul

La complexité de calcul de la version classique de l'algorithme FNTF, les algorithmes proposés dans les sections précédentes, l'algorithme NLMS et l'algorithme du filtre transversal rapide stabilisé FTF8L (8L est pour la complexité) sont énumérés dans le Tableau III.1, dans ce tableau, les complexités de calcul inclut seulement les multiplications et les divisions. Les additions ne sont pas considérées parce qu'elles sont généralement associées aux multiplications dans l'algorithme, ainsi, elles peuvent être incluses dans des opérations de multiplications-accumulations, qui n'ajoutent pas de temps en comparaison avec des multiplications pures.

Le tableau III.1 montre que l'algorithme 2 et l'algorithme 3 ont la même complexité de calcul que la version classique. On peut également voir que l'algorithme 1 a une division supplémentaire par rapport à la version originale de l'algorithme FNTF, cette division est provoquée par l'introduction du facteur gain dans la partie de filtrage de l'algorithme. Cette

modification augmente également la complexité de calcul par P divisions dans l'algorithmes 4 et 5, en raison de l'introduction des P paramètres accélérateurs (une division supplémentaire dans chaque subdivision).

On note également que l'algorithme FNTF classique et ses versions proposées ont une faible complexité en comparaison avec celle de l'algorithme FTF8L quand N est faible, et dans ce cas, leur complexité devient très proche de celle de l'algorithme NLMS.

Algorithmes	Complexité de calcul
classical <i>FNTF</i> algorithm	$2L+12N$ mult.
Algorithme 1	$2L+12N$ mult.+1div
Algorithme 2, Algorithme 3	$2L+12N$ mult.
Algorithm 4, Algorithm 5	$2L+12N$ mult. + P div
Algorithme <i>FTF8L</i> Stabilisé	$8L$ mult.+3 div
Algorithme <i>NLMS</i>	$2L$ mult. + 1div

Tableau III.1: Complexité de calcul de la version originale de l'algorithme FNTF, les algorithmes proposés, l'algorithme NLMS et l'algorithme FTF8L.

III.5. Initialisation des algorithmes proposés

Les conditions initiales de l'algorithme 1 sont les mêmes qu'avec l'algorithme FNTF classique, car la modification est introduite seulement dans la partie de filtrage de la version classique de l'algorithme FNTF. Pour l'initialisation, on suppose que le signal d'entrée est nulle en début de la récurrence ($t=0$) sauf pour $x_{-L} = \sqrt{E_0}$ (initialisation du pré-fenetrage), ou E_0 est l'estimation *a priori* de la puissance d'entrée. Cependant, on suppose que les predicteurs et le gain de Kalman dual sont des vecteurs nulles.

Sous ces suppositions, on trouve que :

$$\alpha_{L,t} = \Gamma_{0,t}^a \text{ et}$$

$$\beta_{L,t} = \Gamma_{0,t}^b ;$$

pour $t \leq 0$, et l'initialisation est réalisée comme décrit dans [4].

$$\Gamma_{0,t}^a = \sum_{i=-\infty}^t \lambda^{t-i} x_i^2 = \lambda^{t+L} E_0 \quad (\text{III.19})$$

$$\Gamma_{0,t}^b = \sum_{i=-\infty}^t \lambda^{t-i} x_{i-N}^2 = \lambda^{t+L-N} E_0 \quad (\text{III.20})$$

Et nous obtenons finalement les conditions initiales suivantes:

$$\alpha_{L,0} = \lambda^L E_0 ,$$

$$\alpha_{L,-L+N} = \lambda^N E_0 ,$$

$$\begin{aligned}\beta_{L,0} &= \lambda^{L-N} E_0, \\ \beta_{L,-L+N} &= E_0,\end{aligned}$$

Les initialisations des algorithmes proposés 2, 3, 4 et 5 ne sont pas très différentes de celle de l'algorithme 1. Elles sont seulement une répétition de l'initialisation globale de l'algorithme classique FNTF, c'est-à-dire, nous avons appliqué les mêmes conditions initiales pour chaque P subdivisions comme suite:

Tous les prédicteurs $\mathbf{a}_{N_i,0}$, $\mathbf{b}_{N_i,0}$, $\mathbf{a}_{N_i,-L_i+N_i}$, $\mathbf{b}_{N_i,-L_i+N_i}$ des parties de prédiction de chaque algorithme proposé et les gains de Kalman duaux $\tilde{\mathbf{c}}_{N_i,0}$, $\tilde{\mathbf{c}}_{N_i,-L_i+N_i}$, $\tilde{\mathbf{c}}_{L_i,0}$ sont supposés nulles, tandis que les variables de vraisemblances $\gamma_{N_i,0}$, $\gamma_{N_i,-L_i+N_i}$, $\gamma_{L_i,t}$ sont initialisées à 1. L'initialisation des variances des erreurs de prédictions sont données par:

$$\begin{aligned}\alpha_{L_i,0} &= \lambda^{L_i} E_0, \\ \beta_{L_i,0} &= \lambda^{L_i-N_i}, \\ \alpha_{L_i,-L_i+N_i} &= \lambda^{N_i} E_0, \\ \beta_{L_i,-L_i+N_i} &= E_0.\end{aligned}$$

Comme nous avons vu dans la section de la stabilisation numérique et également dans cette section, les techniques proposées ne changent pas le contenu de la structure originale de l'algorithme FNTF (lorsque $\mu=0$ et $P=1$), et l'analyse sur la stabilisation et les conditions initiales est maintenue les mêmes que dans l'algorithme FNTF classique. Nous notons également que la constante d'initialisation est importante seulement dans la phase transitoire de chaque algorithme, et selon les valeurs minimales de E_0 que l'algorithme atteint l'optimum après quelques itérations.

La phase permanente de chaque algorithme dépend seulement des valeurs de λ , mais la convergence rapide et les propriétés de la capacité de poursuite de tous les algorithmes proposés ont besoin des valeurs faibles de λ . Ce dernier dépend de la méthode de stabilisation numérique utilisée (contraintes).

III.6. Analyse de la vitesse de convergence et la capacité de poursuite des cinq versions algorithmiques proposées

Dans cette section, nous allons présenter les analyses que nous avons faites sur la capacité de poursuite et sur la vitesse de convergence des cinq nouvelles versions algorithmiques de la version originale du FNTF. Avant de commencer nos analyses, nous allons définir certains critères et donner certaines définitions sur l'algorithme FNTF classique et sur la méthodologie et le modèle de performance analytique qu'on suit dans ce paragraphe.

III.6.1. Modèles de performance

On suppose que le signal désiré y_t est donné par:

$$y_t = \mathbf{h}_{opt}^T \mathbf{x}_{L,t} + e_{opt,t} \quad (\text{III.21})$$

ou \mathbf{h}_{opt} représente le vrai system à identifié et e_{opt} est le bruit additif à la sortie avec une variance v_{min} , indépendant avec le signal d'entrée x_t . On définit le vecteur erreur comme suite :

$$\mathbf{D}_{N,t} = \mathbf{h}_{L,t} - \mathbf{h}_{opt},$$

La constante de temps d'un algorithme adaptative $\mathbf{h}_{L,t}$ est définit par le temps (en nombre d'itérations) que l'algorithme met pour atteindre l'optimum, c'est à dire :

$$\begin{aligned} \mathbf{h}_{L,t} &= \mathbf{h}_{opt}, \\ \tau^{FNTF} &= \frac{1}{1-\lambda} \end{aligned} \quad (\text{III.22})$$

Le disadjustemnt mesure l'exactitude de la convergence. Dans le contexte non stationnaire, on doit ajouter une contribution du retard au disadjustement [52]. On considère un modèle additif simple des erreurs quadratiques moyennes (MSE) pour l'algorithme FNTF après la convergence initiale:

$$E(\bar{\epsilon}_{L,t}^2) - v_{min} = v_{min}(M_{adap,\lambda} + M_{lag,\lambda}) \quad (\text{III.23})$$

Ou $M_{adap,\lambda}$ représente le disadjustement due à la puissance du bruit de sortie v_{min} ; et $M_{lag,\lambda}$ est la contribution au disadjustement globale due aux variations temporelles du trajet d'écho. Toutes ces quantités sont normalisées par v_{min} et sont fonctions du taux d'oubli suivant:

$$\psi = 1 - \lambda \quad (\text{III.24})$$

il a été montré dans [52] que si λ est choisi proche de 1, on obtient:

$$M_{adap,\lambda}^{RLS} = \frac{(1-\lambda)L}{1+\lambda} \approx L\psi/2 \quad (\text{III.25})$$

il a été montré aussi que la dérive de disadjustemnt $M_{lag,\lambda}^{RLS}$ est dépendante du type de non stationnarité [53].

III.6.2. Analyse de la vitesse de convergence

III.6.2.1. Algorithme 1

en utilisant les équations de y_t , la variable de vraisemblance, le gain de Kalman et finalement l'expression de l'algorithme proposé (III.11) (l'équation de mise à jour de l'algorithme 1), on trouve la formule suivante simplifiée du vecteur d'erreur $\mathbf{D}_{L,t}$.

$$\mathbf{D}_{L,t} = (\mathbf{I}_L - \mu^{-1}(\mathbf{I}_L - (1-\mu)(\mathbf{I}_L - \lambda\mu\mathbf{R}_{L,t}^{-1}\mathbf{R}_{L,t-1})^{-1}))\mathbf{D}_{L,t-1} + \frac{\mathbf{R}_{L,t}^{-1}\mathbf{x}_{L,t}}{1-\mu\gamma_{L,t}}e_{opt,t} \quad (\text{III.26})$$

On suppose que :

$$\mathbf{R}_{L,t}^{-1}\mathbf{R}_{L,t} = \mathbf{I}_L,$$

cette approximation est valide asymptotiquement pour un signal d'entrée stationnaire x_t et pour un facteur d'oubli exponentiel proche de 1. C'est la raison pour laquelle nous avons proposé l'algorithme 1. En utilisant ces approximations, l'équation (III.26) devient:

$$\mathbf{D}_{L,t} \approx \lambda_h \mathbf{D}_{L,t-1} + \frac{\mathbf{R}_{L,t}^{-1} \mathbf{x}_{L,t}}{1 - \mu \gamma_{L,t}} e_{opt} \quad (\text{III.27})$$

et l'expression du paramètre λ_h devient comme suite:

$$\lambda_h = \frac{(1 - \mu)\lambda}{1 - \lambda\mu} = \frac{\psi\lambda}{1 - \lambda\mu} \quad (\text{III.28})$$

En prenant les valeurs de l'équation (III.27) et en utilisant l'hypothèse de l'indépendance entre $\mathbf{x}_{L,t}$ et e_{opt} , on trouve:

$$E(\mathbf{D}_{L,t}) = \lambda_h E(\mathbf{D}_{L,t-1}) = \lambda_h^t E(\mathbf{D}_{L,0}) \quad (\text{III.29})$$

A partir de (III.29), on note que $E(\mathbf{D}_{N,t})$ décroît exponentiellement à zéro pour $0 \leq \lambda_h < 1$ et $\mu \leq 1$.

La constante de temps de l'algorithme 1 est obtenue par l'utilisation du nouveau facteur d'oubli implicite λ_h qui est donné par (III.28) dans l'équation (III.22) et on obtient:

$$\tau_h = \frac{1}{1 - \lambda_h} = \frac{1 - \mu\lambda}{\psi} = \tau^{FNTF} (1 - \mu\lambda). \quad (\text{III.30})$$

Cette équation (III.30) montre un des grands avantages du nouvel algorithme 1, le choix d'un facteur d'oubli proche de 1 ne limite pas la vitesse de convergence en raison de l'introduction du paramètre μ qui permet de l'ajuster.

On note que le choix d'une valeur négative du paramètre μ conduit à une constante de temps τ_h plus grande que celle de l'algorithme FNTF classique. Pour améliorer la vitesse de convergence de l'algorithme FNTF, on prend les valeurs de μ comprises entre 0 et 1 ($0 < \mu \leq 1$).

III.6.2.2. Algorithmes 3 et 5

Comme nous l'avons expliqué dans la section 2, seulement les algorithmes 3 et 5 qui permettent une amélioration de la vitesse de convergence de l'algorithme FNTF classique. Selon la structure de ces algorithmes, ces deux algorithmes propagent plusieurs algorithmes du FNTF classique. Le but est de les rendre fonctionnels avec des facteurs d'oubli plus petit que leurs valeurs originales. Les constantes de temps des deux algorithmes sont données comme suite:

$$\tau_i^{(3)} = \frac{1}{1 - \lambda_i} \quad (\text{III.31})$$

$$\tau_i^{(5)} = \frac{1 - \mu\lambda_i}{\psi_i} = \tau_i^{FNTF} (1 - \mu\lambda_i) \quad (\text{III.32})$$

ou $i = 1, 2, 3, \dots, P$, et P représente le nombre de subdivisions dans chaque algorithme (on suppose que les longueurs des filtres sont égales et $L_i = L/P$ est un entier).

Les équations (III.31) et (III.32) montrent bien que les nouvelles expressions des constantes de temps des algorithmes 3 et 5 sont nettement plus petites que celles de l'équation (III.22), ceci est une preuve que les algorithmes 3 et 5 améliorent la vitesse de convergence, parce qu'ils permettent d'avoir de faibles valeurs des constantes de temps (améliorent la vitesse de convergence). On note aussi qu'il est facile de montrer que la constante de temps de l'algorithme 5 est plus petite que celle de l'algorithme 3, ceci est bien illustré par l'équation (III.33).

$$\tau_i^{(5)} = \tau_i^{(3)}(1 - \mu\lambda_i) \quad (\text{III.33})$$

III.6.3. Analyse de la capacité de poursuite

III.6.3.1. Algorithme 1

Pour l'analyse de la capacité de poursuite, on suppose que le système est variable dans le temps, c'est-à-dire $\mathbf{h}_{opt} = \mathbf{h}_{opt,t}$, et suit une équation différentielle de second ordre [35]:

$$\mathbf{h}_{opt,t} = \theta \mathbf{h}_{opt,t-1} + \mathbf{W}_t \quad (\text{III.34})$$

Où $|\theta| < 1$ et \mathbf{W}_t est le vecteur du bruit blanc de moyenne nulle et de matrice de covariance égale à $\sigma_W^2 \mathbf{I}_L$.

Le vecteur d'erreur de poursuite $\mathbf{D}_{L,t}$ devient comme suite:

$$\mathbf{D}_{L,t} = (\mathbf{I}_L - \mathbf{M}_{L,t})\mathbf{D}_{L,t-1} + \mathbf{F}_{L,t}e_{opt} + (1 - \theta)\mathbf{h}_{opt,t-1} - \mathbf{W}_t \quad (\text{III.35})$$

où:

$$\mathbf{F}_{L,t} = \frac{\mathbf{R}_{L,t}^{-1} \mathbf{x}_{L,t}}{1 - \mu + \mu \mathbf{x}_{L,t}^T \mathbf{R}_{L,t}^{-1} \mathbf{x}_{L,t}} \quad (\text{III.36})$$

$$\mathbf{M}_{L,t} = \frac{\mathbf{R}_{L,t}^{-1} \mathbf{x}_{L,t} (\mathbf{x}_{L,t} \frac{\mu}{1 - \mu})^T \mu^{-1}}{1 + (\frac{\mu}{1 - \mu} \mathbf{x}_{L,t})^T \mathbf{R}_{L,t}^{-1} \mathbf{x}_{L,t}} \quad (\text{III.37})$$

L'erreur quadratique moyenne donnée par l'équation (III.23) est calculée comme suite:

$$MSE_{ex} = \text{trace}[E(\mathbf{D}_{L,t-1} \mathbf{D}_{L,t-1}^T) \mathbf{R}_{L,xx}] \quad (\text{III.38})$$

où:

$$\mathbf{R}_{L,xx} = E(\mathbf{x}_{L,t} \mathbf{x}_{L,t}^T)$$

À partir de (III.35), avec la supposition $1/(1 - \theta) \gg 1/(1 - \lambda_h)$ et en négligeant le terme $(1 - \theta)\mathbf{h}_{opt}$, on obtient:

$$E(\mathbf{D}_{L,t}\mathbf{D}_{L,t}^T) = E[(\mathbf{I}_L - \mathbf{M}_{L,t})\mathbf{D}_{L,t-1}\mathbf{D}_{L,t-1}^T(\mathbf{I}_L - \mathbf{M}_{L,t}^T) + E[\mathbf{M}_{L,t} \frac{\mathbf{R}_{L,t}^{-1}}{1 - \mu\gamma_{L,t}}]v_{min} + \mathbf{I}_L\sigma_w^2] \quad (\text{III.39})$$

Pour simplifier (III.39), on utilise la technique du lemme d'inversion matriciel, ensuite l'approximation que $\mathbf{M}_{L,t} \approx (1 - \lambda_h)\mathbf{I}_L$. En plus on remplace les variables $\mathbf{R}_{L,t}$ et $\gamma_{L,t}$ par leurs valeurs asymptotiques (pour λ proche de 1):

$$\mathbf{R}_{L,\infty} \approx (1 - \lambda)\mathbf{R}_{L,xx},$$

$$\gamma_{L,\infty} = \lambda^L,$$

Cette dernière approximation découle de la définition suivante:

$$\gamma_{L,t} = \lambda^L \frac{\det(\mathbf{R}_{L,t-1})}{\det(\mathbf{R}_{L,t})} = \lambda^L \frac{\beta_{L,t}}{\alpha_{L,t}} \quad (\text{III.40})$$

Où $\beta_{L,t}$ et $\alpha_{L,t}$ dénotent respectivement, les variances des erreurs de prédictions aller et retour. En utilisant ces approximations, l'équation (III.39) devient:

$$E(\mathbf{D}_{L,t}\mathbf{D}_{L,t}^T) = \lambda_h^2 E(\mathbf{D}_{L,t-1}\mathbf{D}_{L,t-1}^T) + \frac{(1 - \lambda_h)(1 - \lambda)}{1 - \mu\lambda^L} \mathbf{R}_{L,xx}^{-1} v_{min} + \mathbf{I}_L\sigma_w^2 \quad (\text{III.41})$$

Cette équation contient deux récurrences: la première (qui dépend de v_{min}) donne $M_{adap,\lambda}$ et la deuxième (qui dépend de σ_w^2) donne $M_{lag,\lambda}$

En itérant (III.41) de $t=0$ et en utilisant (III.23), on obtient:

$$M_{adap,\lambda} = \frac{(1 - \lambda)}{(1 + \lambda_h)} \frac{L}{1 - \mu\lambda^L} v_{min} \quad (\text{III.42})$$

$$M_{lag,\lambda} = \frac{1}{1 - \lambda_h^2} L \sigma_x^2 \sigma_w^2 \quad (\text{III.43})$$

On note que si $\mu=0$ et $\lambda_h=\lambda$, on obtient les formules de $M_{lag,\lambda}$ et $M_{adap,\lambda}$ dans la cas de l'algorithme FNTF classique. A partir de l'équation (III.42) on observe que $M_{adap,\lambda}$ atteint sa valeur maximale (égale à v_{min}) pour $\mu=1$ et pour $\lambda^L = 1 - (1 - \lambda)L$. Mais le résultat le plus important est que ceci est vraie pour $0 < \mu \leq 1$, $\lambda_h < \lambda$; ceci montre que $M_{lag,\lambda}$ de l'algorithme 1 est plus faible que le $M_{lag,\lambda}$ de la version classique de l'algorithme FNTF.

Enfin, on note que pour $\mu < 0$, ($\lambda_h > \lambda$), l'algorithme 1 est plus lent que la version classique de l'algorithme FNTF.

III.6.3.2. Algorithmes 2, 3, 4 et 5

L'analyse que nous avons faite pour l'algorithme 1 est également vraie pour chaque algorithme proposé mais avec des arrangements spéciaux pour quelques paramètres importants qui forment les équations (III.42) et (III.43) comme μ , λ et λ_h . Par exemple, pour l'algorithme 2 et 3, on met $\mu = 0$ et $\lambda_h = \lambda_i$ (λ_i est le facteur d'oubli de chaque subdivision ayant une longueur de filtre égale à $L_i = L/P$, ou P représente le nombre de subdivisions temporelles utilisées dans la conception de l'algorithme en question).

Pour les algorithmes 4 et 5 qui sont les combinaisons entre l'algorithme 1 et les algorithmes 2 et 3, respectivement, nous pouvons obtenir leurs performances par les réglages suivants:

$$\begin{aligned} \mu &= \mu_i, \\ \lambda_h &= \lambda_{hi} \\ \text{et } \lambda &= \lambda_i. \end{aligned}$$

On note que chaque algorithme proposé mène à des capacités de poursuites plus grandes que celle de la version originale de l'algorithme FNTF, et tous ces algorithmes fonctionnent avec des valeurs du facteur d'oubli plus petites que celle de l'algorithme FNTF classique. Ceci est l'origine de l'amélioration de leurs performances. En conclusion, les 5 nouveaux algorithmes ne changent pas le contenu de l'algorithme FNTF, mais utilisent des techniques qui leur permettent de surmonter les contraintes du facteur d'oubli en réduisant implicitement sa valeur et sans altérer les propriétés numériques de l'algorithme FNTF classique.

III.7. Résultats expérimentaux des nouvelles versions de l'algorithme FNTF

III.7.1. Test de la vitesse de convergence

Afin d'évaluer la performance de la vitesse de convergence des algorithmes proposés et de la version originale du FNTF dans des cas stationnaires, un canal acoustique prélevé (échantillonné) à 16 kHz dans une voiture et qui est composé de 256 coefficients a été utilisé. Le canal acoustique a été mesuré dans des conditions réelles [42,] (Figure III.3). Le signal d'entrée est un signal stationnaire (bruit blanc stationnaire). Pour évaluer la performance de la vitesse de convergence de chaque algorithme FNTF, le critère suivant a été utilisé:

$$J_t = 10 \text{Log}_{10} \left(\frac{\langle \bar{\varepsilon}_{L,t}^2 \rangle}{\langle y_{L,t}^2 \rangle} \right) \quad (\text{III.44})$$

Où $\langle * \rangle$ représente un moyennage temporelle à court terme sur 265 échantillons.

Les figures III.4, III.5 et III.6 montrent, respectivement, la performance de convergence des algorithmes 1, 3 et 5. Pour ces figures, les algorithmes proposés ont été employés pour identifier une réponse impulsionnelle finie ($L=256$). Un saut dans la réponse d'impulsion a été simulé au bloc 90 (1 bloc = 256 échantillons), qui correspond à 1.44ms à la fréquence d'échantillonnage $F_s=16$ kHz, afin d'évaluer la vitesse de convergence en phase permanente. Toutes ces versions sont comparées à l'algorithme original du FNTF.

On peut voir que l'algorithme 1 améliore la vitesse de convergence dans les deux régimes, transitoire et permanent (c'est-à-dire, au début et après le bloc 90). L'amélioration est plus importante dans la phase permanente pour les algorithmes 3 et 5, avec une légère dégradation des valeurs finales du NMSE (normalized mean square error).

D'après ces simulations, on remarque une faible performance des algorithmes 3 et 5 dans la phase transitoire. Ceci est dû à la convergence initiale de l'algorithme FNTF qui dépend de la constante initiale E_0 et du facteur d'oubli λ_i de chaque subdivision d'ordre N_i (les variances des erreurs de prédiction aller et retour α_{N_i} et β_{N_i} pour chaque subdivision sont initialisées par E_0).

Si E_0 est petite en comparaison avec la puissance du signal d'entrée, les premières valeurs de $\gamma_{N,t}$ sont faibles et les nouveaux algorithmes 1,3 et 5 deviennent équivalents à l'algorithme FNTF classique. En revanche, pour des grandes valeurs de t , la vitesse de convergence dépend seulement du facteur d'oubli, et Les algorithmes modifiés 1, 3 et 5 ont une vitesse de convergence nettement meilleure que celle de l'algorithme original. Ceci est clairement montré sur les figures III.4, III.5 et III.6, pour chaque nouvel algorithme.

Quant aux algorithmes 2 et 4, ils ne rapportent aucune amélioration de la vitesse de convergence (courbes non montrées), toutefois ils permettent une amélioration significative de la capacité de poursuite quand le système acoustique varie dans le temps, comme sera montré dans le paragraphe suivant.

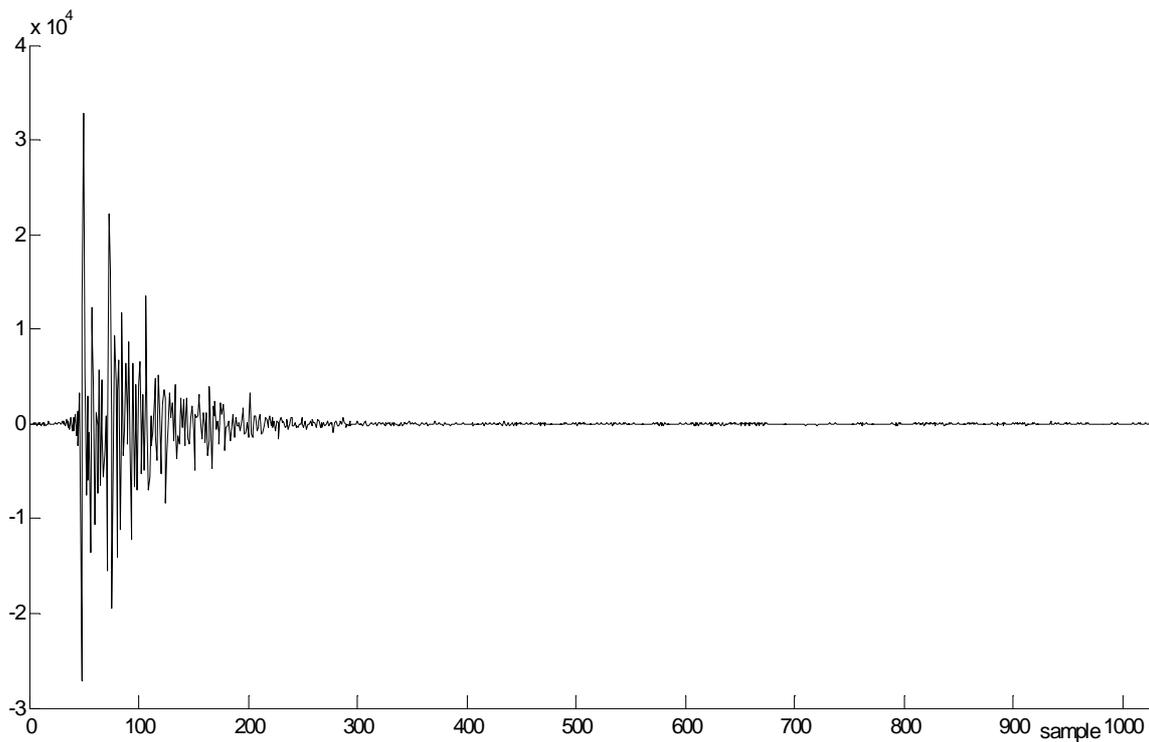


Figure III.3. Réponse impulsionnelle du canal acoustique mesurée dans une voiture (échelles verticales est une unités arbitraires).

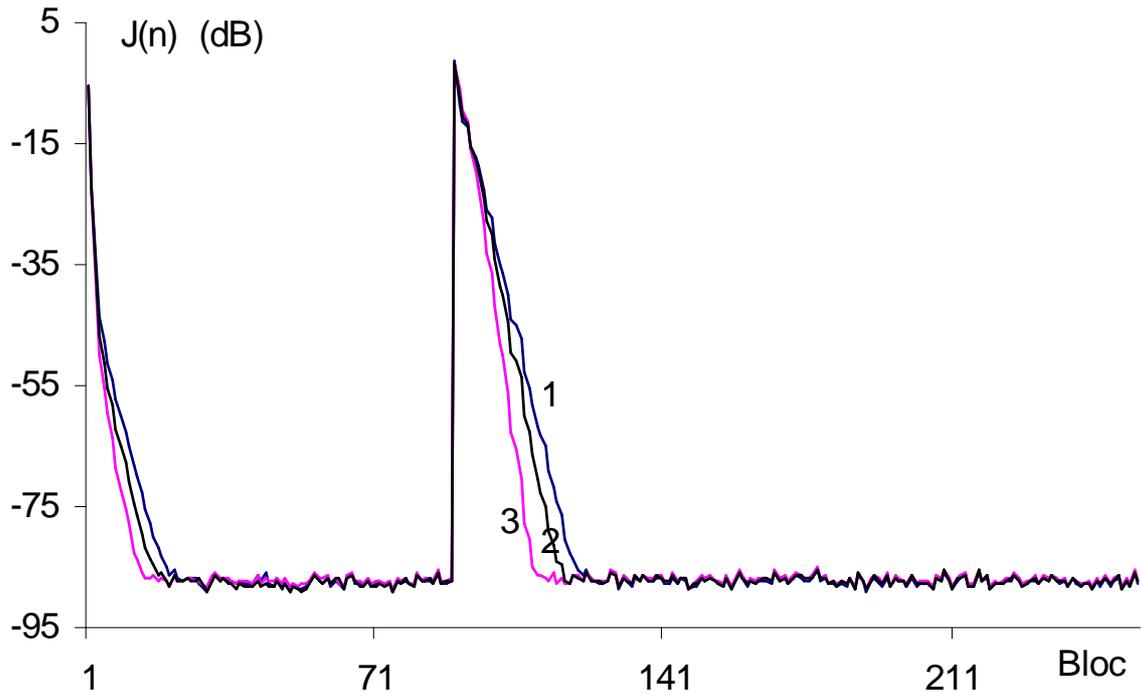


Figure III.4. performance de convergence de l'algorithme1, signal d'entrée est du bruit blanc.
 (1): $\mu=0$ (*FNTF*); (2): $\mu=0.5$; (3): $\mu=0.998$.

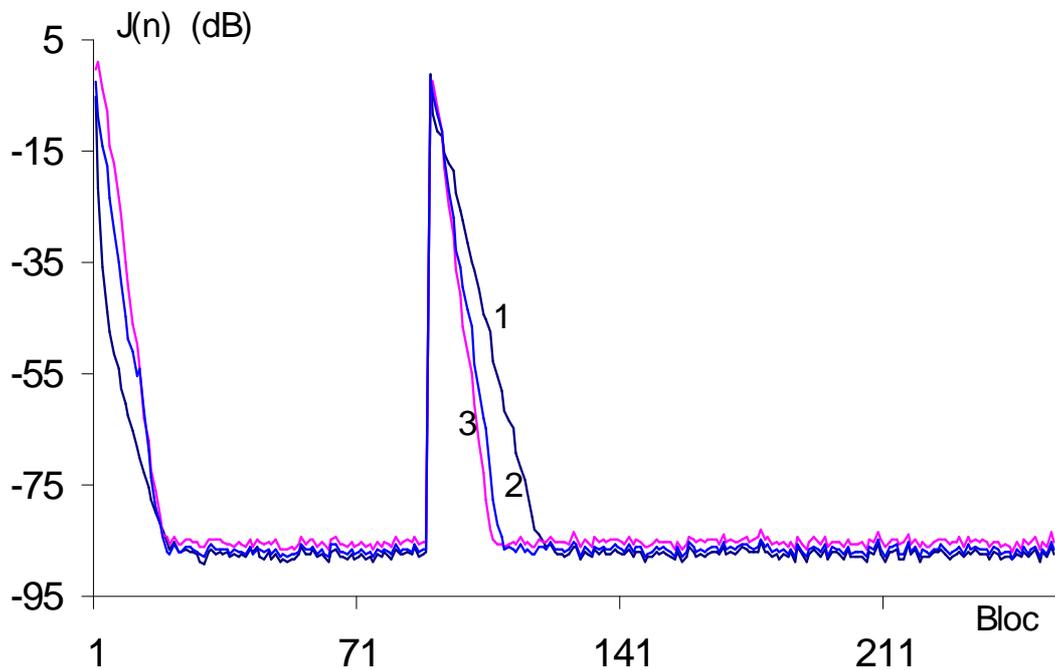


Figure III.5. : performance de convergence de l'algorithme3, signal d'entrée est du bruit blanc.
 for $P=1$ (*FNTF*), $P=3$ and $P=4$ subdivisions.

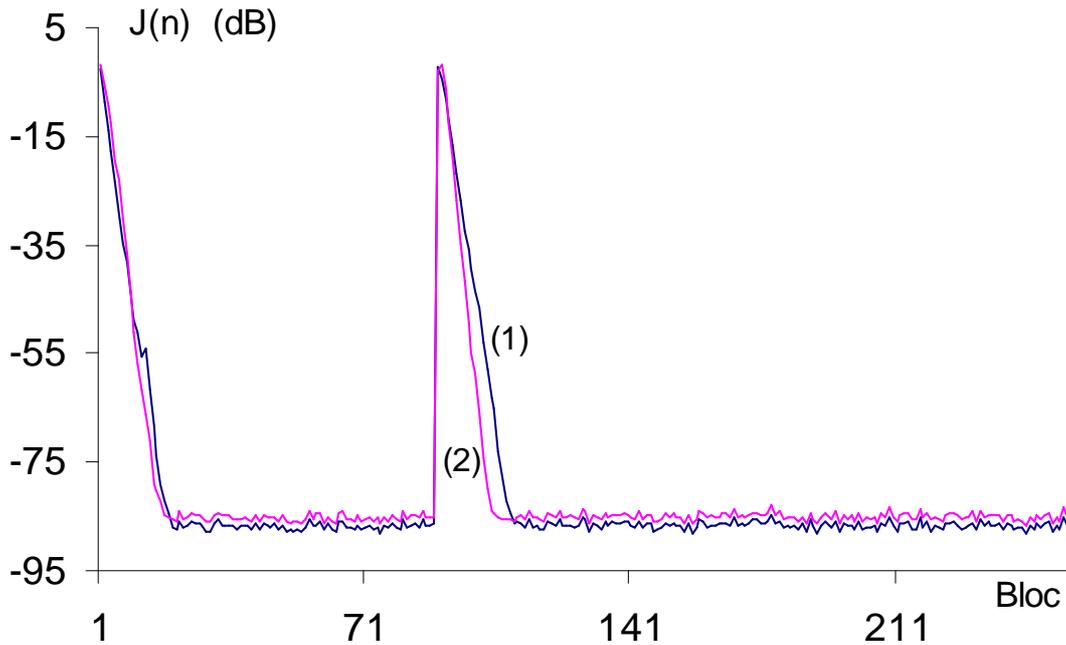


Figure III.6. : performance de convergence de l'algorithme 5, signal d'entrée est du bruit blanc.
 (1): Algorithm 3 with $P=2$; (2): Algorithm 5 with $P=2$ and $\mu=0.5$

III.7.2. Test de la capacité de poursuite avec un canal variable dans le temps

Pour tester la capacité de poursuite des algorithmes FNTF, nous avons utilisé un canal acoustique variable dans le temps [42,57,58]. Les variations temporelles de l'écho acoustique ont été produites par un déplacement lent d'une personne entre le microphone et le haut-parleur dans le système d'installation.

Le signal d'entrée est un bruit stationnaire de spectre moyen de la parole, il est échantillonné à 16kHz. Ce canal est estimé par un filtre transversal de 1700 coefficients (Figure III.7). Dans ce cas aussi, le même critère a été utilisé pour évaluer la propriété de poursuite de chaque algorithme FNTF.

Les figures III.8, III.9 et III.10 comparent l'identification et la poursuite du vrai trajet de l'écho acoustique par les algorithmes proposés 1, 2 et 3 et par l'algorithme classique FNTF, pour plusieurs cas de μ et de P . Ces figures montrent clairement les meilleures capacités de poursuite de chaque nouvelle version algorithmique du FNTF par rapport à la version originale. On peut voir que l'amélioration de la performance en capacité de poursuite devient plus importante avec de grandes valeurs de P ; si la valeur du paramètre P augmente, le facteur d'oubli pour chaque subdivision diminue et la vitesse de convergence augmente en conséquence. Le même effet a été noté avec l'algorithme 1, avec l'introduction du paramètre μ qui permet une amélioration significative quand il est choisi proche de 1.

On a noté aussi que l'utilisation de l'erreur de filtrage finale pour actualiser les coefficients des filtres locaux dans l'Algorithme 3 permet une amélioration importante de la capacité de poursuite par rapport à l'algorithme 2.

Les figures III.11 et III.12 correspondent aux algorithmes 4 et 5 respectivement. On observe que la combinaison des techniques employées par ces 2 algorithmes mène à une amélioration significative de la capacité de poursuite par rapport à la version classique du FNTF et les algorithmes 2 et 3 (c'est-à-dire sans le paramètre accélérateur μ).

Ces résultats montrent bien l'efficacité de combiner les deux techniques (c'est-à-dire, modification de la partie de filtrage et sa subdivision temporelle). Quoique chaque nouvel algorithme améliore la capacité de poursuite de la version originale du FNTF, on remarque aussi que l'amélioration de la capacité de poursuite de l'algorithme 5 est plus meilleure que les autres algorithmes, car dans l'algorithme 5, on a utilisé la technique du découpage temporelle plus l'erreur de filtrage finale pour l'actualisation des filtres locaux, donc il rassemble les avantages de leurs bons dispositifs.

On devrait également noter qu'aucun problème numérique de divergence n'a été expérimenté dans les simulations [36,47,59].

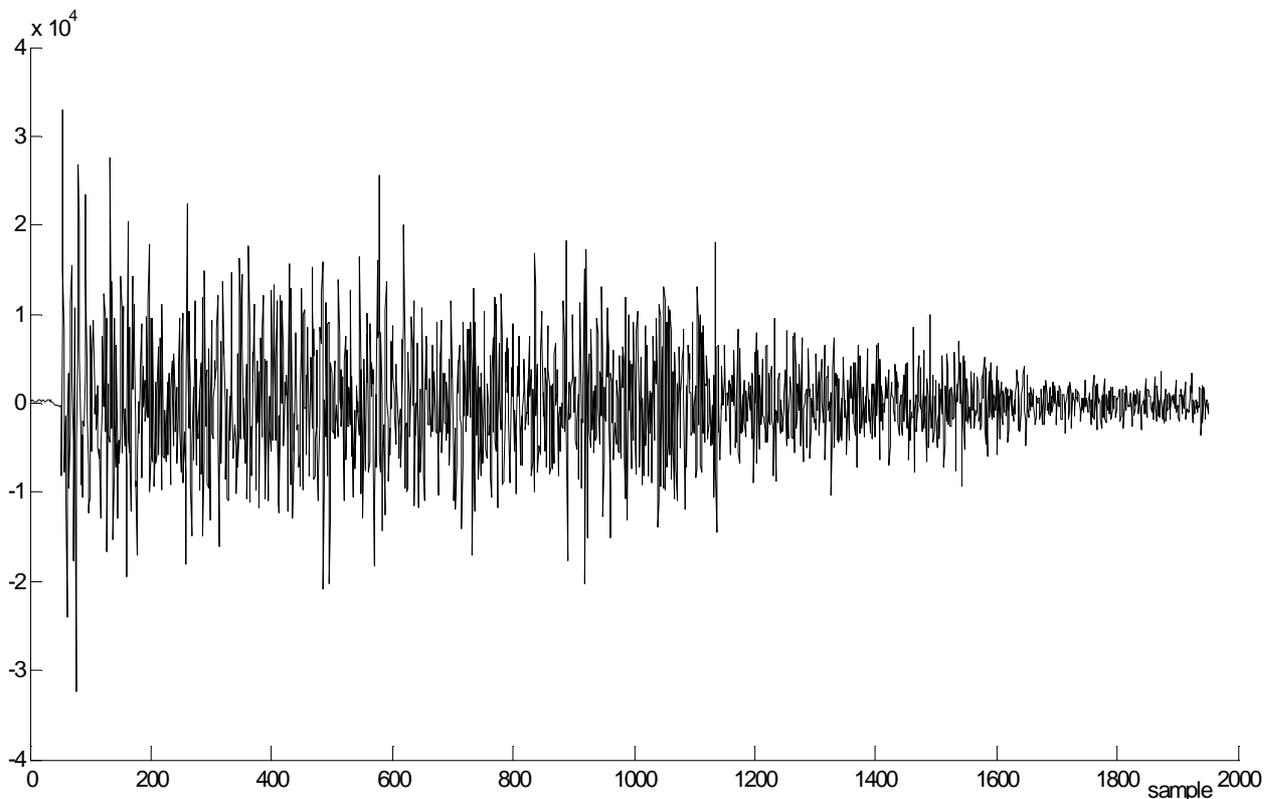


Figure III.7: Réponse impulsionnelle de l'écho acoustique mesurée dans une salle (échelle verticale avec unité arbitraire)

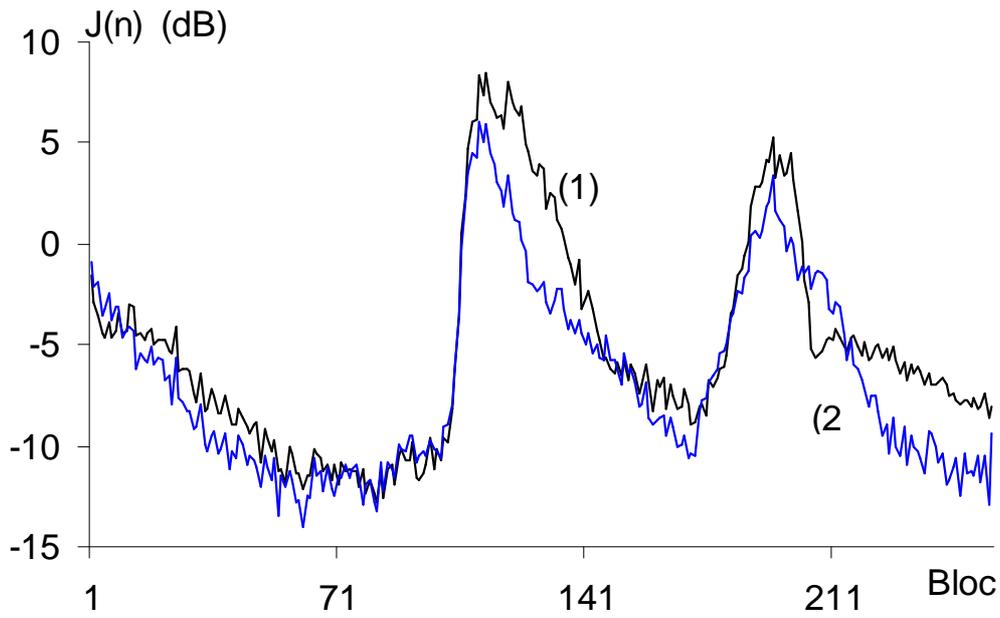


Figure III.8. Performance en convergence et en capacité de poursuite de l'Algorithme 1.
 (1): $\mu=0$ (FNTF); (2): $\mu=0.92$.

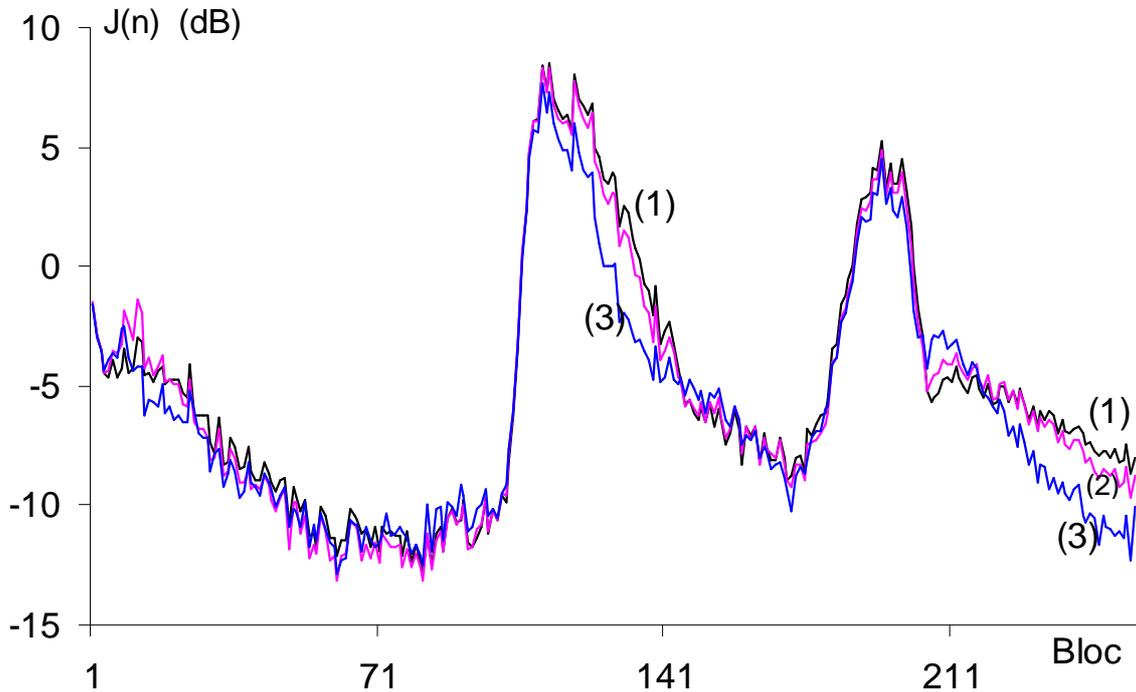


Figure III.9. Performance en convergence et en capacité de poursuite de l'Algorithme 2.
 (1): $P=1$ (FNTF); (2): $P=2$; (3): $P=3$.

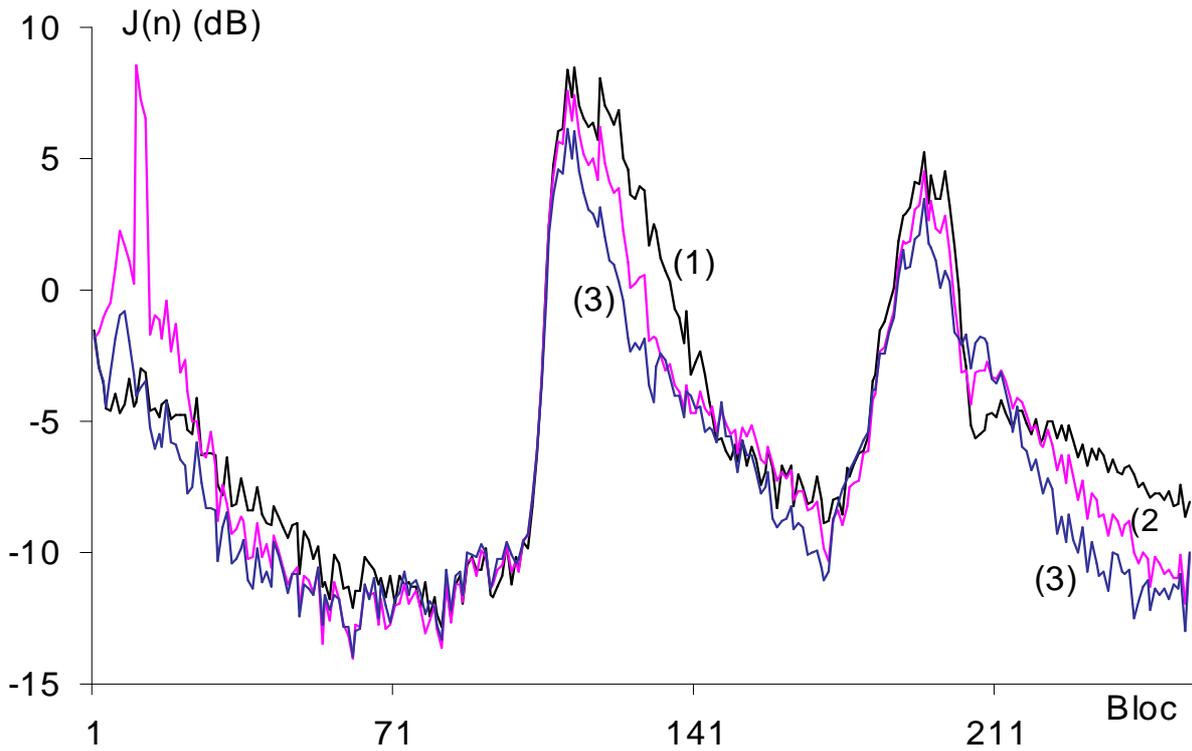


Figure III.10. Performance en convergence et en capacité de poursuite de l'Algorithme 3.
 (1) $P=1$ (FNTF); (2): $P=2$; (3): $P=3$.

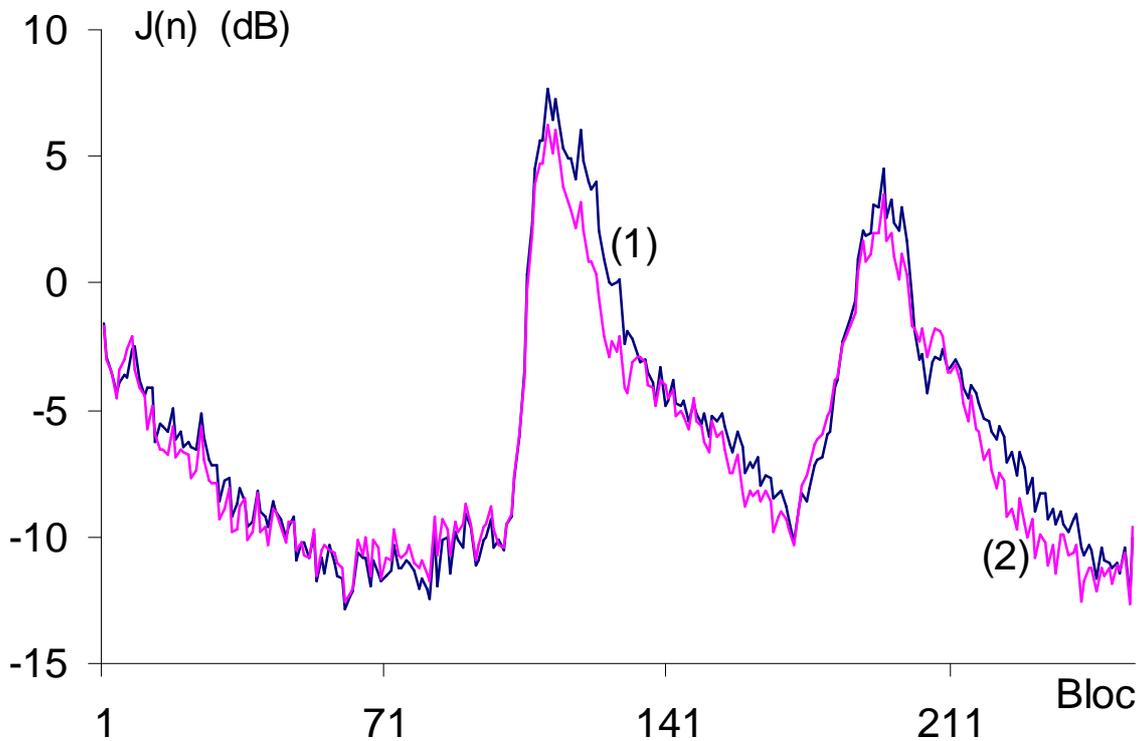


Figure III.11. : Performance en convergence et en capacité de poursuite de l'Algorithme 4.
 (1): Algorithm 2 with $P=3$ and $\mu=0$; (3): Algorithm 4 with $P=3$ and $\mu=0.5$

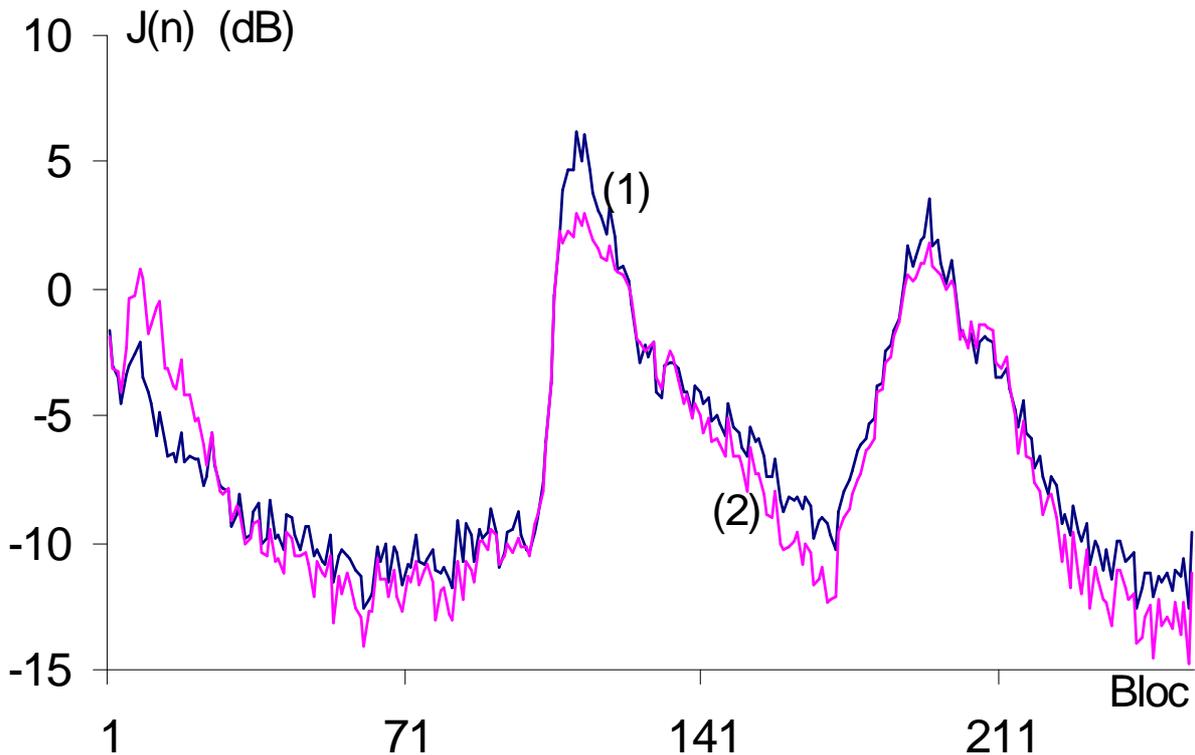


Figure III.12. : Performance en convergence et en capacité de poursuite de l'Algorithme 5.

(1): Algorithm 3 with $P=3$ and $\mu=0$; (2): Algorithm 5 with $P=3$ and $\mu=0.65$.

III.7.3. Test de la capacité de poursuite avec un canal variable dans le temps artificiellement

Dans cette expérience, nous avons simulé des variations du trajet d'écho réel en variant le gain à la sortie du canal (canal temporellement varié artificiellement). Le canal est le même que précédemment (voir Fig.III.3), suivi d'une commande de gain. Nous avons choisi des taux de variations qui ont produits des erreurs du même ordre que des erreurs dues à des variations réels, et les variations du canal sont réalisées entre le bloc 125 et 190 (2ms à 3.04ms).

Le signal d'entrée est un bruit de spectre moyen de la parole sans pic dominant; la taille totale L du filtre est 256; le canal est formé des 256 premiers échantillons d'une réponse impulsionnelle d'une voiture (système sans troncation); et le SNR à la sortie est (RSB) est 60dB (bruit blanc de sortie).

La figure III.13, montre trois courbes obtenues à partir de l'algorithme 1. La courbe 1 est celle de l'algorithme FNTF classique et les courbes 2 et 3 sont obtenues à partir de l'algorithme 1 avec $\mu = 0,5$ et $0,95$ respectivement. Nous notons que l'algorithme 1 avec $\mu = 0,95$ est meilleur en poursuite (il est en dessous de la courbe 1 de -10dB et de -5dB de la courbe 2). Ceci prouve que le taux de poursuite de l'algorithme FNTF traditionnel est amélioré avec l'introduction du paramètre μ . On note aussi que les réglages de l'algorithme 1 ($\mu = 0,5$ et $0,95$) ont pratiquement la même vitesse initiale de convergence. Mais en régime permanent, la vitesse de convergence de l'algorithme FNTF avec $\mu = 0,95$ devient la meilleure.

De la figure III.14, nous notons que l'algorithme 3 permet une nette amélioration de la capacité de poursuite de l'algorithme FNTF classique. Cette amélioration est due à la réduction de la valeur du facteur d'oubli, grâce à la division temporelle de la structure 2. Le niveau des erreurs de poursuite de l'algorithme 3 avec $P=4$ est en dessous des erreurs de l'algorithme FNTF de -13dB . On note également que cette version d'algorithme a une faible vitesse de convergence en phase initiale. Cet inconvénient devient moins significatif dans le régime permanent.

Trois courbes sont montrées sur la figure III.15: la courbe 1 représente la version classique de l'algorithme FNTF, la deuxième courbe représente l'algorithme 3 avec $P=4$ et la dernière courbe est celle de l'algorithme 5 avec $P=4$ et $\mu=0,6$. Nous pouvons clairement observer la supériorité en performance de la capacité de poursuite de l'algorithme 5 par rapport les deux autres algorithmes (les erreurs de poursuite de l'algorithme 5 sont au-dessous de celle de l'algorithme 3 de -10dB), ceci est dû au fait que l'algorithme 5 rassemble les avantages des autres algorithmes (division temporelle plus une accélération dans la partie de filtrage). Par conséquent, on note que l'introduction d'un paramètre accélérateur dans la partie de filtrage de l'algorithme 5 améliore sa vitesse de convergence en phase initiale, mais dans cette phase, la version classique de l'algorithme FNTF reste la meilleure (vitesse de convergence en phase initiale).

Pour les algorithmes 2 et 4, et comme nous l'avons expliqué auparavant, il n'y a aucune amélioration de la vitesse de convergence de l'algorithme FNTF classique. Cependant, ils permettent une amélioration de la capacité de poursuite de ce dernier (FNTF classique).

III.7.4. Comparaison des performances avec un canal acoustique variable dans le temps

Dans les expériences rapportées dans cette section, nous avons utilisé le même signal d'entrée et le même canal comme dans la section précédente. Nous avons également utilisé la même technique de contrôle de gain pour faire varier le canal dans le temps. Pour la comparaison des algorithmes, le pas d'adaptation de l'algorithme NLMS est pris égale à 1 et qui donne la meilleure capacité de poursuite dans le cas de notre étude. Le facteur d'oubli de l'algorithme FTF8L est pris égale à :

$$\lambda = 1 - \frac{1}{3L + 3.5},$$

Où L est la longueur du filtre.

Pour les autres algorithmes nous avons choisi les paramètres qui ont donné la meilleure capacité de poursuite selon les résultats des simulations obtenus dans les sous-sections (III.7.2) et III.7.3). Pour l'algorithme 1, nous avons pris $\mu=0,95$, pour l'algorithme 3 nous avons pris $P=4$, et pour l'algorithme 5 nous avons choisi $P=4$ et $\mu=0,95$ (qui est la combinaison des deux algorithmes). Sur les figures III.16, III.17 et III.18, nous n'avons pas montré la courbe de l'algorithme FNTF classique parce qu'il a donné les mêmes résultats que l'algorithme FTF8L, qui est pris comme référence dans chaque expérience.

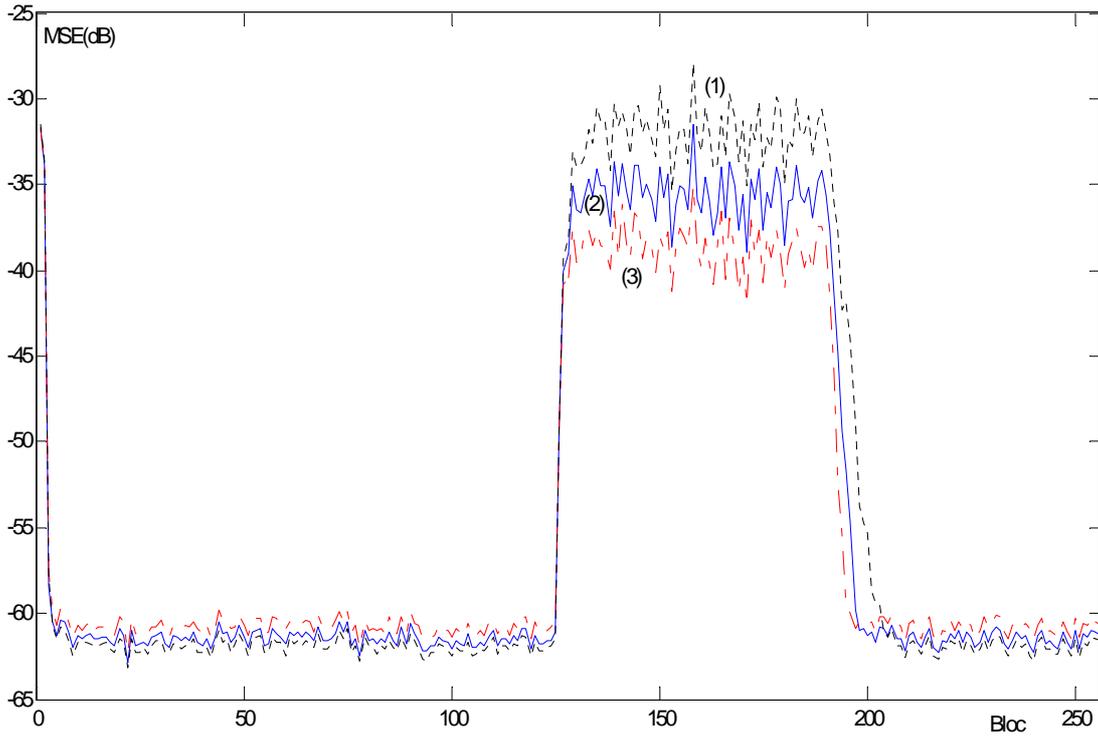


Figure III.13. Performance de poursuite de l'Algorithme 1,
 (1): $\mu=0$, (2): $\mu=0.5$, (3): $\mu=0.9$.

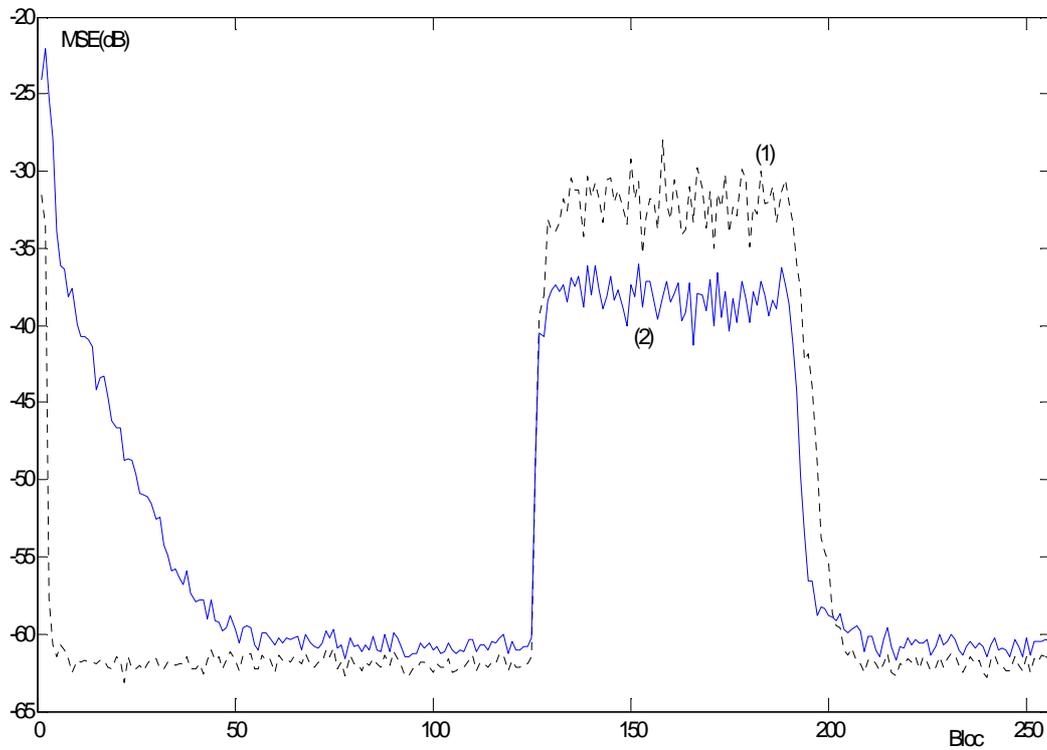


Figure III.14. Performance de poursuite de l'Algorithme 3,
 (1): $P=1$ (FNTF); (2): $P=4$.

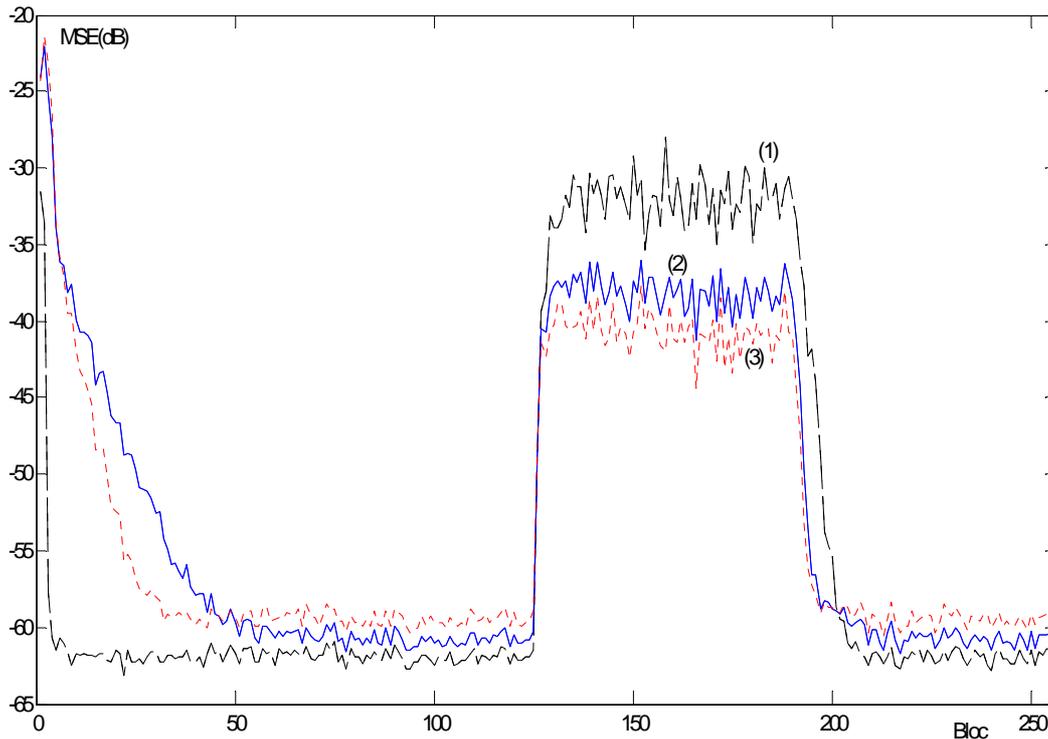


Figure III.15: Performance de la vitesse de convergence et la poursuite de l'Algorithme 5, (1): $P=1$ and $\mu=0$, (2): $P=4$ and $\mu=0$, (3) $P=4$ and $\mu=0.6$

De la figure III.16, nous notons que l'introduction du paramètre accélérateur μ dans la partie de filtrage a donné de meilleure performance pour l'algorithme 1, car cette modification lui permet de fonctionner avec un facteur d'oubli implicite plus faible que celui de la partie de prédiction, et comme nous le savons, accélérer la poursuite dans les algorithmes des moindres carrés est l'équivalente à la diminution du facteur d'oubli λ . Cependant, l'algorithme NLMS est connu avec ses bonnes propriétés de poursuite. Ceci est bien montré sur la figure III.16.

L'algorithme 1 montre une bonne performance comparée à l'algorithme FTF8L et il a presque les mêmes propriétés en poursuite que l'algorithme NLMS, qui prouve l'efficacité de l'algorithme modifié. Nous notons également que l'algorithme NLMS montre un comportement faible dans la phase transitoire et dans le régime permanent. Ceci est dû à sa dépendance aux statistiques du signal d'entrée.

Sur la figure III.17, une comparaison similaire est faite avec l'algorithme 3. Nous notons que l'algorithme 3 avec $P=4$ a approximativement les mêmes erreurs de poursuites que l'algorithme 1 avec un $\mu = 0,95$, et le comportement de l'algorithme 3 en phase initiale est très proche de celui du NLMS, mais sa vitesse de convergence s'améliore dans le régime permanent après l'arrêt de la variation du canal.

Nous concluons que cet algorithme se comporte mieux quand les coefficients du filtre sont identifiés par lui même et nous pouvons obtenir les mêmes propriétés que l'algorithme 1 avec le paramètre $P=4$. En outre cet algorithme montre ses bonnes propriétés devant l'algorithme FTF8L, qui a les mêmes propriétés que l'algorithme FNTF classique.

La figure III.18 montre un résultat très intéressant de l'algorithme 5. Nous notons que cet algorithme a une meilleure vitesse de convergence par rapport l'algorithme 3 et le NLMS, et il a la meilleure performance en poursuite par rapport les autres algorithmes.

Ces résultats prouvent l'efficacité du mixage de l'algorithme 1 avec l'algorithme 3. Nous notons également la mauvaise performance de l'algorithme NLMS en phase transitoire et même dans la période après l'arrêt de la variation du canal. Nous concluons que l'algorithme 5 offre une meilleure performance en comparaison avec celle de l'algorithme NLMS qui est connu par ses bonnes propriétés de poursuite quand le canal est variable dans le temps, en outre, nous avons noté que l'algorithme FTF8L (le même que l'algorithme FNTF) a un comportement faible avec le canal variable dans le temps et nous avons proposés des algorithmes qui améliorent son comportement.

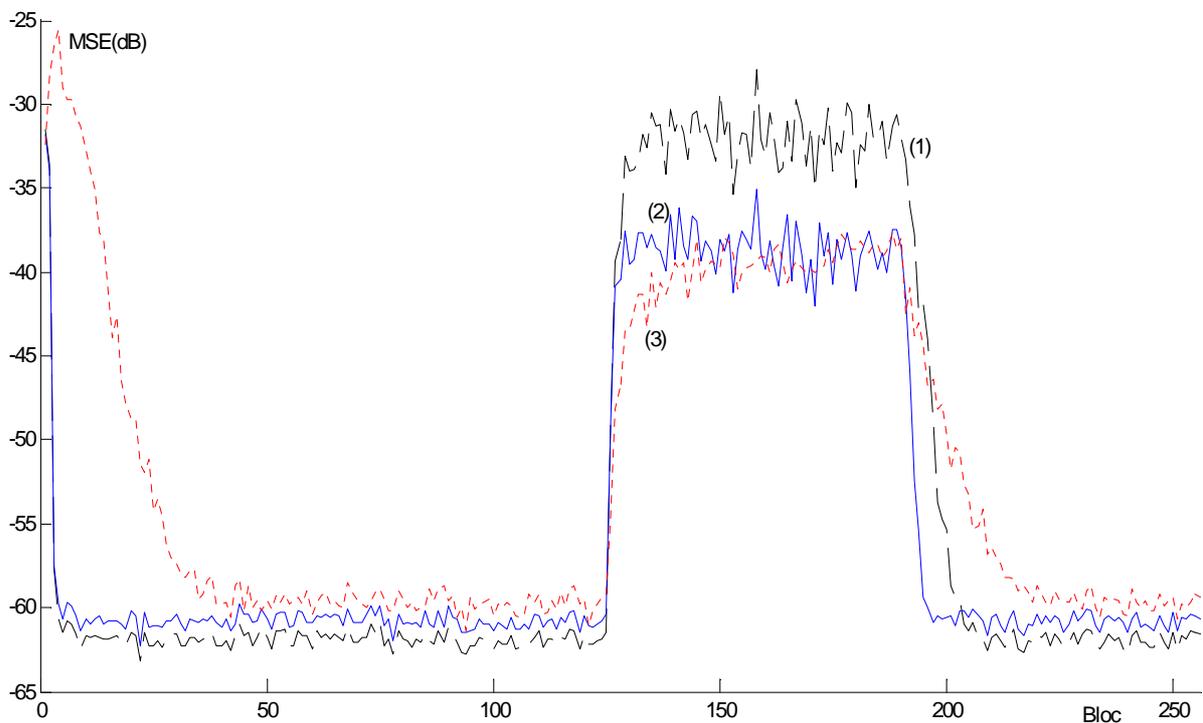


Figure III.16: Comparaison des performances des algorithmes:
(1): *FTF8L*, (2): *Algorithm 1* and (3): *NLMS*.

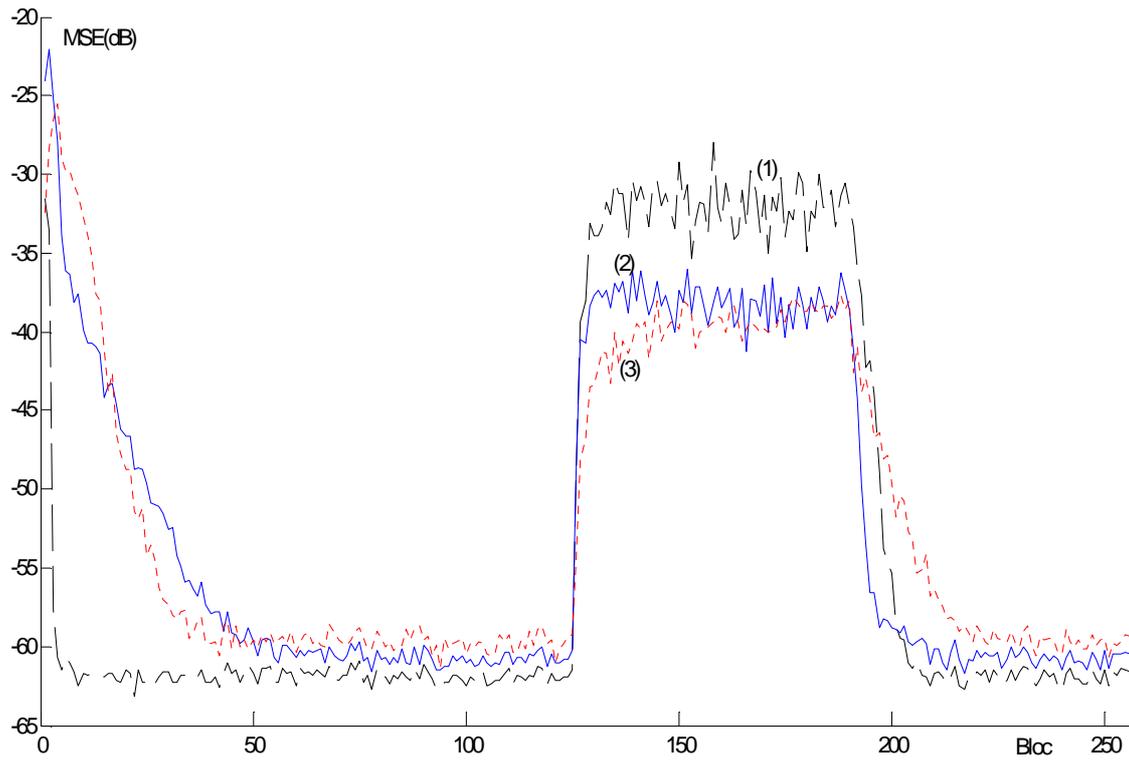


Figure III.17: Comparaison des performances des algorithmes :
 (1): *FTF8L*, (2): *Algorithm 3* and (3): *NLMS*.

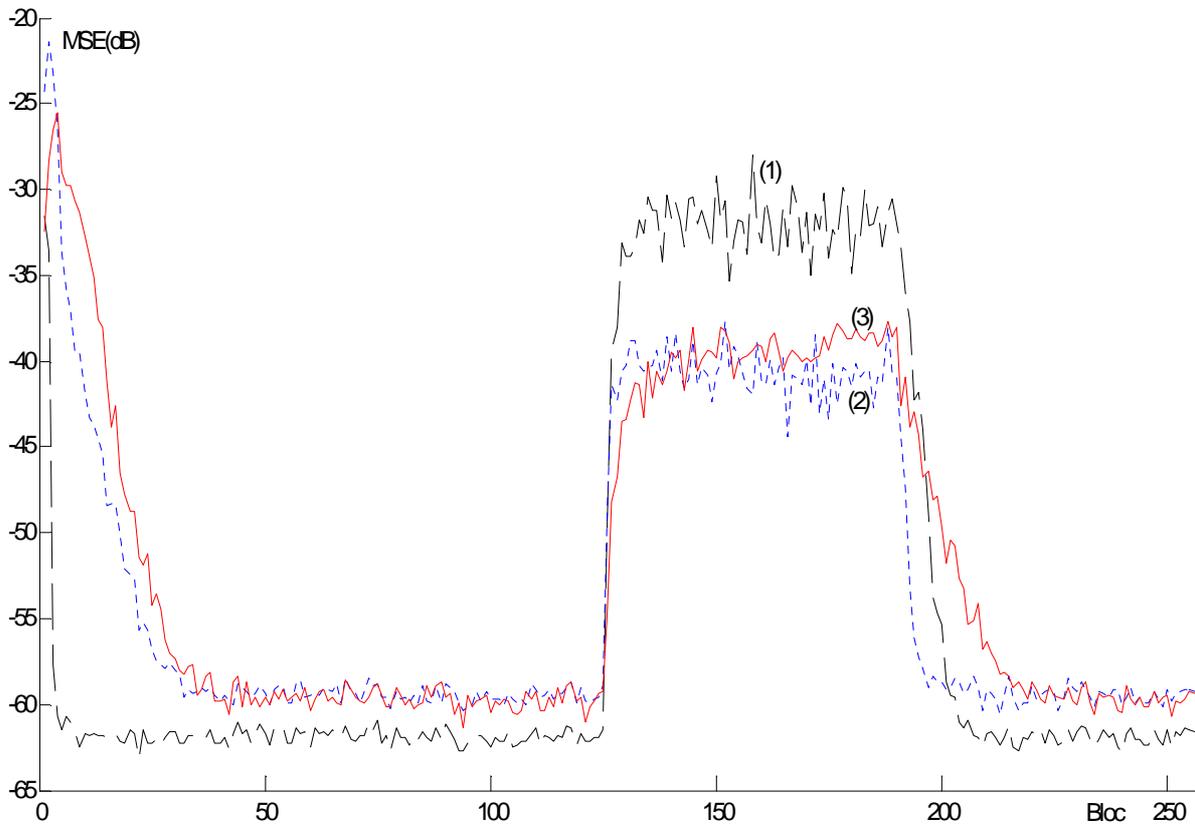


Figure III.18: Comparaison de performances des algorithmes :
 (1): *FTF8L*, (2): *Algorithm 5* and (3): *NLMS*.

III.8. Algorithme FNS par bloc et simulations

Plusieurs familles d'algorithmes par blocs ont été développées avec comme principal objectif la réduction de la complexité des calculs. Le principe de base de ces algorithmes par blocs est l'actualisation du filtre adaptatif qui est effectuée une fois par bloc de taille M échantillons.

Nous présentons dans ce paragraphe une nouvelle versions par bloc de l'algorithme de Newton rapide simplifié [9].

L'algorithme Bloc-FNS est construit à partir du FNS en imposant que les coefficients du filtres adaptatifs restent fixes sur chaque bloc de taille M . On définit la matrice d'échantillons (de taille $M \times L$) :

$$\underline{X}_t = [X_{t-M+1} \quad X_{t-M+2} \quad \dots \quad X_t] \quad (\text{III.45})$$

Le filtre H étant maintenu fixe entre les instants $t-M+1$ et t , le vecteur des M erreurs successives (différentes de celles du NLMS) est donné par :

$$\tilde{E}_t = [\tilde{\epsilon}_{t-M+1} \quad \tilde{\epsilon}_{t-M+2} \quad \dots \quad \tilde{\epsilon}_t]^T \quad (\text{III.46})$$

satisfait à :

$$\tilde{E}_t = Y_t - \underline{X}_t^T H_{t-M} \quad (\text{III.47})$$

avec :

$$Y_t = [y_{t-M+1} \quad y_{t-M+2} \quad \dots \quad y_t]^T. \quad (\text{III.48})$$

On impose le pas d'adaptation μ_B fixe sur tout le bloc, l'équation d'adaptation par bloc du filtre s'écrit :

$$H_t = H_{t-M} + \mu_B \tilde{E}_t \tilde{C}_{N,t} \quad (\text{III.49})$$

De la même manière que le bloc-FNS, l'adaptation des coefficients du filtre adaptatif se fait une fois par bloc de taille M échantillons consécutifs sauf que les équations d'adaptations des trois algorithmes par Bloc diffèrent de celles du Bloc-FNS. Ces équations sont résumées dans le tableau III.2 suivant :[9].

Type d'algorithmes par Bloc	Equations d'adaptations du filtre transverse
Bloc-FTF8L.	$H_t = H_{t-M} + \tilde{C}_{L,t} \tilde{E}_t$ (III.50)
Bloc-FNTF12N2L	$H_t = H_{t-M} + \tilde{C}_{N,t} \tilde{E}_t$ (III.51)
Bloc-FNS	$H_t = H_{t-M} + \mu_B \tilde{E}_t \tilde{C}_{N,t}$ (III.52)

Tableau III.2 : équations d'adaptations du filtre transverse des trois algorithmes par blocs.

avec :

$\tilde{C}_{L,t}$: est le vecteur du gain de Kalman dual d'ordre L.

$\tilde{C}_{N,t}$: est le vecteur du gain de Kalman dual d'ordre N.

μ_B : est le pas d'adaptation de la partie de filtrage de l'algorithme Bloc-FNS.

\tilde{E}_t : est le vecteur des M erreurs successives .

Le tableau III.3 suivant donne la complexité des calculs de ces algorithmes .

Ce tableau montre que ces versions d'algorithmes proposées permettent une importante réduction de la complexité des calculs par rapport à leurs versions originales qu'on peut les retrouver en mettant la taille du bloc M égale à 1.

Type d'algorithmes	Nombre de Multiplications	Nombre de Divisions	Complexité (MADPR)
BFTF8L	$7L+L/M$	3	$7L+L/M+48$
BFTF12N2L	$L +12N+L/M$	7	$L +12N+L/M +112$
BFNS(version :1)	$L +6N+L/M$	0	$L +6N+L/M$
BFNS(version :2)	$L +3N+L/M$	0	$L +3N+L/M$
BNLMS	$L+L/M$	1	$L +16+L/M$

Tableau III.3 : Complexité des calculs des algorithmes:
BFTF8L, BFTF12N2L, BFNS et BNLMS.

Le tableau III.4 suivant donne un exemple de la complexité des calculs relative des algorithmes d'identification adaptative citée au dessus par rapport à celle du NLMS et par rapport leurs versions originales pour une taille du filtre transverse égale à (L-256), un ordre de prédiction égale à (N-16) et une taille du bloc d'échantillons (M-16).

Type d'algorithmes	Complexité. (MADPR)	Par rapport à /NLMS	Amélioration en complexité
BFTF8L	1856	3.625	11.45 %
BFTF12N2L	576	1.125	29.41 %
BFNS(version :1)	368	0.71875	39.47 %
BFNS(version :2)	320	0.625	42.85 %
BNLMS	272	0.53125	48.48 %

Tableau III.4: Complexités des calculs relatives des algorithmes par bloc:
BFTF8L, BFTF12N2L, BFNS et BNLMS.
L=256,N=16, M=16.

Pour mieux comprendre le comportement de l'algorithme BFNS par rapport a l'algorithme FNS nous avons tracé la réponse de ces deux algorithmes pour une entrée d'un signal auto régressif (AR) d'ordre 10 d'un bruit blanc. Le signal x_t est construit par la convolution du bruit blanc par un système tout pôle d'ordre 10. Les pôles de ce système sont :

$$0.98e^{\pm j0.02\pi}, 0.96e^{\pm j0.21\pi}, 0.8e^{\pm 0.96\pi}, 0.75e^{\pm j0.45\pi}, 0.7e^{\pm j0.44\pi}$$

La figure III.19 représente le résultat de la comparaison de cet algorithme avec sa version originale ($M=1$) pour des différentes tailles du bloc M , ($M=4,8,16$). Ce résultat montre que la dégradation de la vitesse de convergence, en régime permanent, augmente avec l'augmentation de la taille du bloc M .

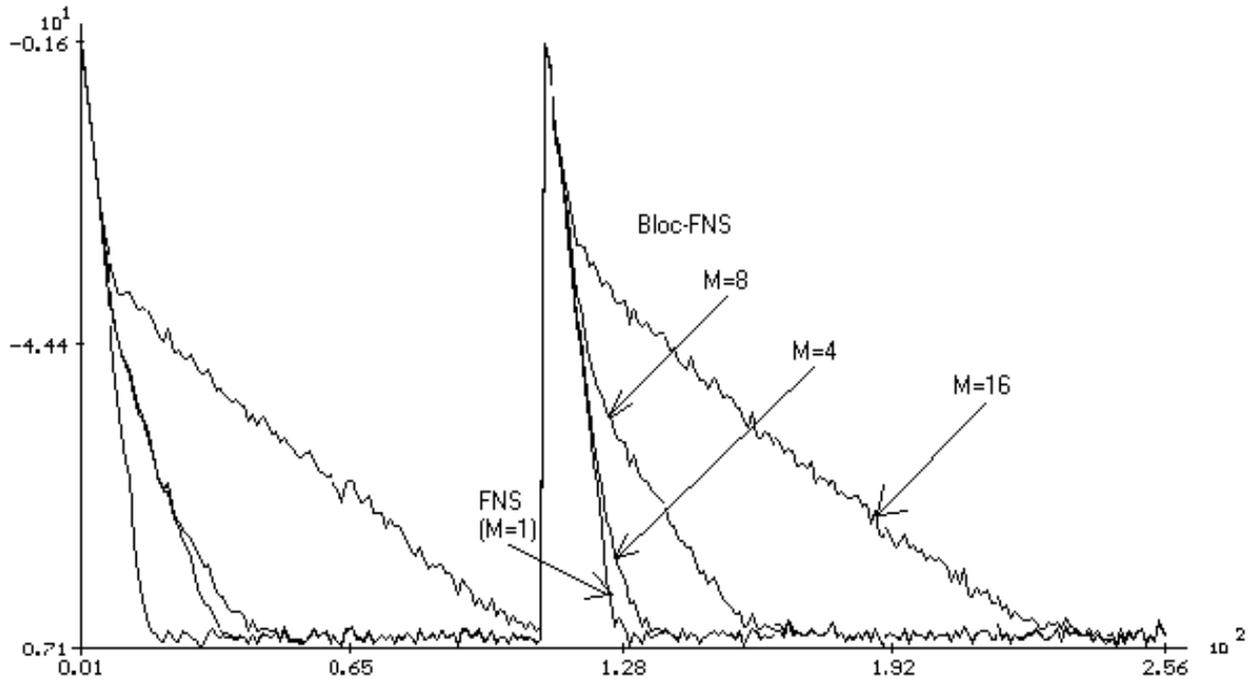


Figure III.19 : Comparaison de la vitesse de convergence de l'algorithme BFNS avec sa version originale FNS($M=1$), pour $M=4,8,16$ signal d'entrée: un bruit blanc. $L = 256$ et $N = 16$.

III.9. Nouvelles méthodes combinatoires de l'algorithme FTF

Comme nous l'avons expliqué dans le chapitre I et II, l'algorithme FTF est numériquement stable sous certaines conditions. La source de son instabilité est due à la partie de prédiction et à l'accumulation des erreurs numériques dans le calcul des variables de prédiction rétrograde. Nous avons mentionné aussi que cet algorithme a une seule partie de prédiction et le fonctionnement de cette partie est conditionné par la contraintes sur la méthode de stabilisation numérique utilisée. Nous avons penser au même principe que celui utilisé avec l'algorithme FNTF classique lorsque on propose l'utilisation de l'algorithme 4 et 5 qui sont basés sur des combinaisons entre les techniques du découpage temporelle de la structure de prédiction de base ainsi que l'ajout de la performance de l'introduction du paramètres μ dans la partie de filtrage. Ces deux méthodes combinatoire ont permis de construire deux nouvelles versions de l'algorithme FTF et ils ont donnée pratiquement le même résultat que l'algorithme FNTF lorsque la taille des predcteurs est assez proche de celle de la taille du filtre transverse, les résultats de simulations ne sont pas montrés car elles ressemblent beaucoup à celles de l'algorithme FNTF lorsque N est grande. Les résultats de simulations de ces deux méthodes combinatoire avec l'algorithme FTF sont publié en [55,56].

III.10. Conclusion

Dans ce chapitre, cinq nouvelles versions de l'algorithme du filtre de Newton transversal rapide (FNTF: Fast Newton Transversal Filter) ont été présentées. Trois des algorithmes proposés améliorent la vitesse de convergence pour des cas stationnaires. Les résultats de simulation ont également prouvé que tous les algorithmes proposés du FNTF améliorent la capacité de poursuite

L'étude comparative a également prouvé que l'algorithme 5 est plus efficace que les autres algorithmes, parce qu'il rassemble les avantages des deux techniques, c'est-à-dire, la modification de la partie de filtrage et sa subdivision temporelle.

Deux autres versions de l'algorithme FTF ont été présentées. Ces deux versions algorithmique permettent aussi l'amélioration de la vitesse de convergence et la capacité de poursuite du FTF classique, le principe est le même que celui de l'algorithme 4 et 5 et les résultats sont très identiques à ces deux derniers lorsque les réglages sont faits très proches de l'algorithme FTF classique.

Nous avons montré aussi que la nouvelle version par bloc de l'algorithme FNTF simplifié (FNS) dégrade les performances de l'algorithme originale lorsque la taille du bloc M est importante, mais elle permet une réduction importante de la complexité de calcul.

Comme perspectives à ce travail, l'adaptation dynamique de la variable μ basée sur les propriétés stationnaires ou non stationnaire de l'environnement pourra introduire une amélioration en convergence et en poursuite. Une opération similaire pourra être faite avec le découpage de la partie de prédiction en fonction de la dynamique des variations des systèmes acoustiques dans le temps. Une méthode efficace en annulation d'écho acoustique avec l'algorithme de filtrage adaptatifs qui est basée sur ce principe sera présentée dans le prochain chapitre.

Chapitre IV

Proposition d'une nouvelle méthode pour l'amélioration de l'annulation d'écho acoustique par les algorithmes adaptatifs

IV.1. Introduction

L'annulation d'écho acoustique est un problème qui se pose dans un grand nombre d'applications (téléphonie main libre, téléconférence,.....). Les difficultés essentielles rencontrées sont d'une part liées à la durée des réponses impulsionnelles des canaux acoustiques à identifier et d'autre part à la nature des signaux à traiter. [1]

Une communication téléphonique introduit des effets gênants selon le milieu où l'on se trouve et la distance entre les deux interlocuteurs. Pour remédier à ces inconvénients, les ingénieurs se sont penchés particulièrement sur les problèmes suivants, pour ne citer que les plus importants [2] ;

- Les réverbérations : en plus du son direct, le ou les microphones de prise de son captent une multitude de réflexions sur les parois de l'habitable.
- Les différents bruits de fonds qui existent en l'absence ou en la présence du signal utile (personne se déplaçant dans la salle,).
- Les échos : qui sont particulièrement gênants à partir d'un certain retard dépassant environ 50 ms.

Dans ce chapitre nous allons présenter une nouvelle technique qui permet d'améliorer l'annulation d'écho acoustique dans les systèmes multimédia [60]. Cette méthode est basée sur le contrôle des paramètres de l'algorithme de filtrage adaptatif utilisé dans l'annulation d'écho acoustique en fonction de la nature des tranches constituant le signal de parole. Ce dernier est constitué des zones de silences où l'énergie est presque nulle, des zones du signal voisées et d'autres qui ne le sont pas. Pour les techniques de la segmentation du signal de parole, nous avons utilisé plusieurs techniques et nous en avons retenu deux types de techniques, la technique de segmentation par fenêtres fixe et celle récursive [61,62].

Pour le choix des algorithmes, nous avons opté pour un choix de famille, c'est à dire nous avons choisis deux types de familles d'algorithmes, le premier type est celui du gradient stochastique et le deuxième est le nouveau algorithme de Newton qui a été détaillé dans le chapitre III. Ce choix est motivé par le fait que cet algorithme permet une nette amélioration de la vitesse de convergence ainsi que de la capacité de poursuite et il est plus facile à adapter à la méthode que nous proposons dans ce chapitre. L'élément essentiel qui permet cette adaptation est soit son facteur d'oubli exponentiel ou bien son nouveau paramètre qu'on a introduit dans la partie de filtrage. Les deux

techniques mènent aux mêmes performances et permettent une nette amélioration de l'annulation d'écho acoustique. Les résultats de simulations de trois types d'applications ont montré l'efficacité de cette technique dans le domaine de l'annulation d'écho adaptative.

IV.2. Principe de base de l'annulation d'écho acoustique

L'écho acoustique est gênant pour le correspondant distant dès que le retard de transmission dans le réseau de télécommunication dépasse 30 ms (communication longue distance, transmission par satellite); il doit donc être atténué le plus possible. Pour les tests et les comparaisons, nous nous plaçons dans le cas de l'annulation d'écho acoustique dans un milieu acoustique sonorisé (salle). Cet écho acoustique a pour origine la propagation du son depuis le haut-parleur vers le microphone d'un terminal (poste par exemple « main libre »); cette propagation inclut les réflexions sur les parois de la salle et sur les obstacles présents dans cette salle.

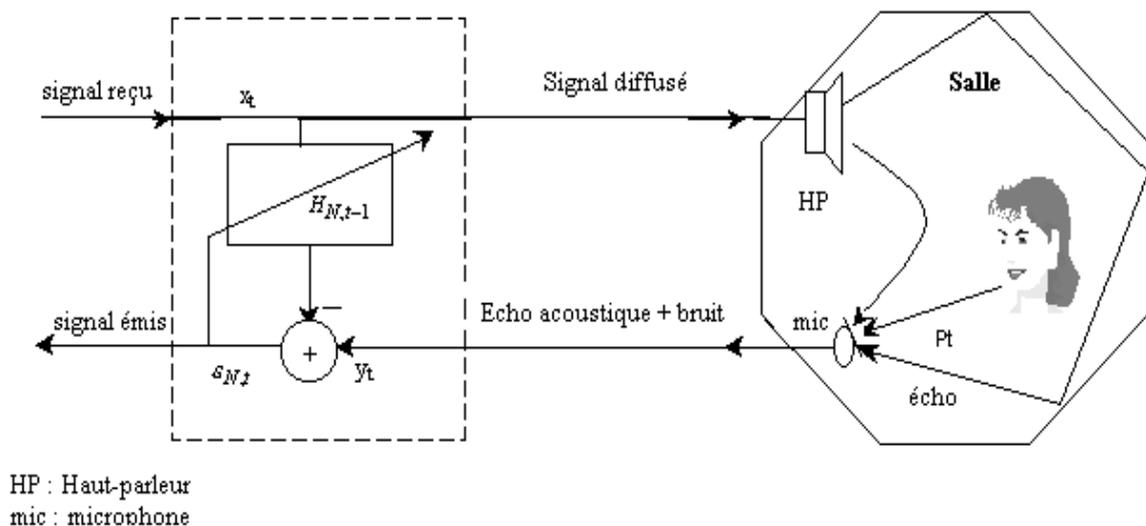


Figure IV.1 : modèle de terminal audio avec l'écho acoustique.
La partie encadrée schématise le principe de l'annulation d'écho.

Le problème qui nous intéresse dans cette étude est l'estimation continue d'un signal s_t , éventuellement perturbé par un autre signal x_t , appelé signal de référence. La composante p_t est non mesurable, elle est incluse dans l'observation captée sur une voie dite principale :

$y_t = s_t + p_t$, x_t est le signal reçu du locuteur lointain, y_t est le signal d'écho du locuteur lointain vers lui-même et p_t est le bruit ambiant du milieu acoustique ou le signal de la double parole du locuteur proche.

La restauration du signal utile est réalisée en soustrayant à l'observation y_t une estimée \hat{s}_t , du signal s_t obtenu par un filtrage approprié du signal de référence x_t [5]. Le schéma de principe de ce filtrage est représenté sur la Figure IV.2 où les signaux sont connus aux instants d'échantillonnage t . La période d'échantillonnage est T_e , et l'on a : $x_t = x(t) = x(t \cdot T_e)$.

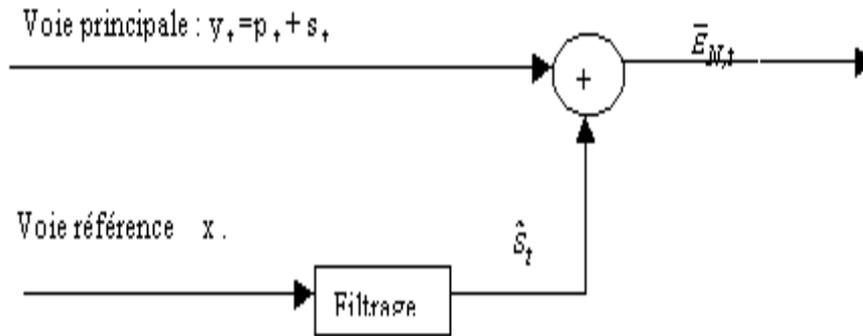


Figure IV.2 : schéma de principe d'un système d'annulation d'écho acoustique

Il faut noter que l'optimum en l'annulation d'écho acoustique est d'avoir la sortie $\mathcal{E}_{N,t} = 0$ en absence de double parole ou du bruit perturbateur ($p_t = 0$).

Les hypothèses de base concernant le système de la figure IV.2 sont d'une part l'indépendance entre la composante p_t et les signaux s_t et x_t , et d'autre part la corrélation des signaux s_t et x_t .

IV.3. Description des critères de performance

Le critère de performance objectif, couramment utilisé en annulation d'écho, est l'évolution temporelle de l'énergie de l'erreur de filtrage normalisée par l'énergie du signal d'écho y_t , exprimée en décibels (c'est l'atténuation de l'énergie de l'écho) :

$$J(t) = 10 \log_{10} \left(\frac{\sigma_{\bar{\mathcal{E}}_{N,t}}^2}{\sigma_{\bar{y}_{N,t}}^2} \right)$$

Où σ^2 désigne une moyenne temporelle effectuée sur un certain nombre d'échantillons consécutifs.

Avec la parole, ce critère de performance privilégie les zones du signal de fortes énergies. Il n'est pas très significatif dans les zones de très faibles énergies (silences) où l'erreur de filtrage et le signal ont des niveaux comparables.

Pour cette raison, nous utilisons, dans la suite, le critère objectif qui donne l'évolution temporelle de l'énergie de l'erreur de filtrage *a priori*, exprimée en dB et donnée par :

$$J(t) = \text{EQM}(t) = 10 \log_{10} \left(\sigma_{\bar{\mathcal{E}}_{N,t}}^2 \right).$$

avec

$$\sigma_{\bar{\mathcal{E}}_{N,t}}^2 = \frac{1}{L} \sum_{i=1}^L \bar{\mathcal{E}}_{N,t}^2.$$

Où $\bar{\mathcal{E}}_{N,t}$ représente l'erreur de filtrage *a priori* issu d'algorithme de filtrage adaptatif en cours d'étude soit (NLMS ou bien l'algorithme FNTF version 1 (Algorithme 1)).

IV.4. Acquisition des réponses impulsionnelles

Dans notre étude, nous nous plaçons dans le cas d'un canal acoustique stationnaire. Pour simuler le milieu acoustique, nous disposons de trois réponses impulsionnelles de couplage acoustique mesurées en régime stationnaire (Figure IV.3). La première est une réponse impulsionnelle mesurée dans un habitacle de voiture, et la deuxième dans une salle de visioconférence et la troisième est celle d'une salle d'audioconférence. Ces trois réponses impulsionnelles contiennent respectivement 1024, 4096, 8192 points [57].

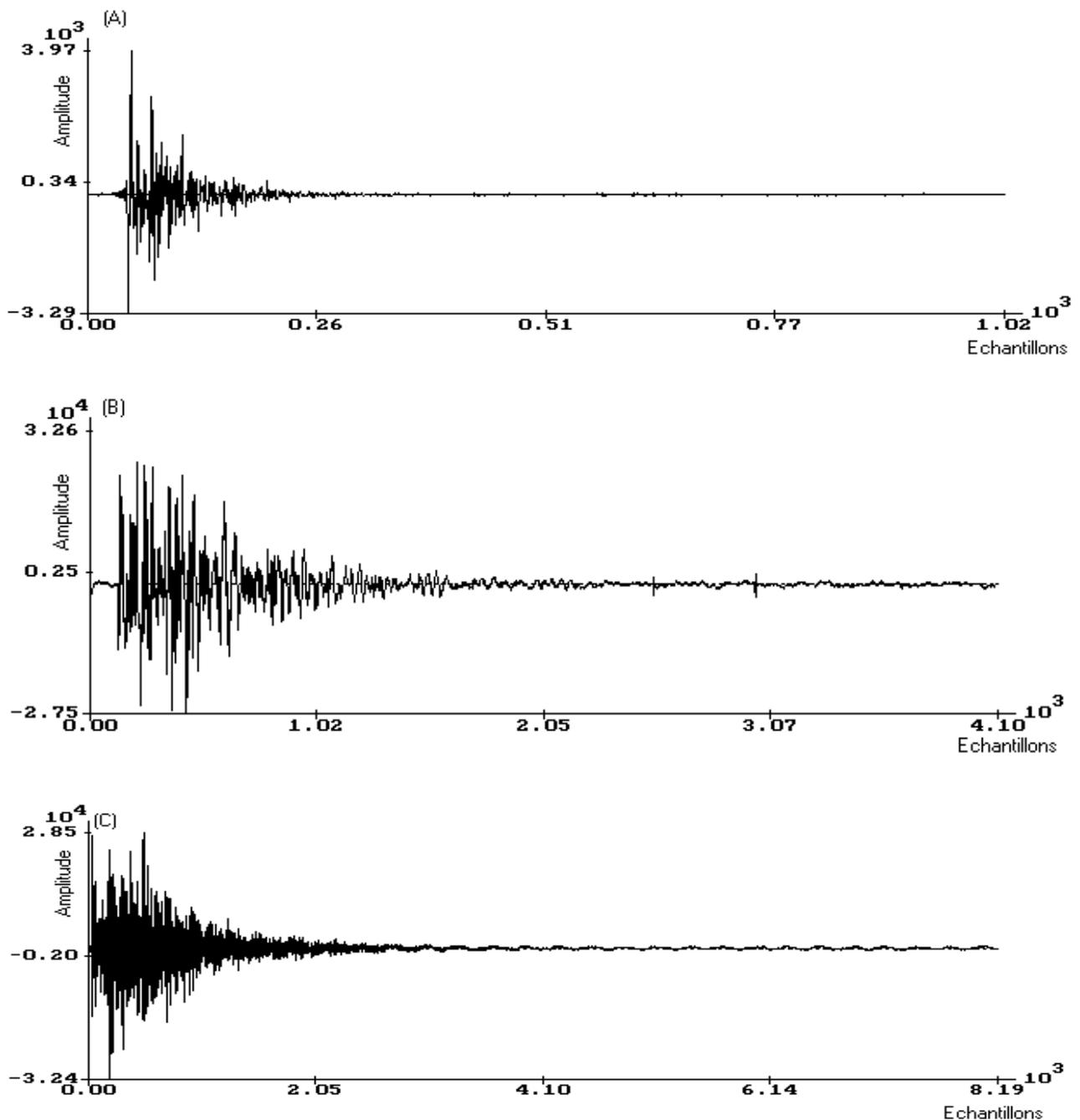


Figure IV.3 : Réponses impulsionnelles de canaux de couplage acoustiques.

- (A) habitacle de voiture.
- (B) salle de visioconférence.
- (C) salle d'audioconférence.

IV.5. Résultats de simulations de la technique proposée

Nous présenterons ici quelques résultats de simulations que nous avons obtenues avec la méthode récursive et celle des fenêtres fixes pour deux types d'algorithmes, le NLMS qui est basé sur le choix du pas d'adaptation μ et la version 1 de l'algorithme FNTF, qui est basée sur le choix du facteur d'oubli exponentiel λ .

IV.5.1. Algorithme NLMS Modifié (NLMSmr)

D'après les résultats de simulations effectués avec l'algorithme NLMS, nous avons constaté le bon comportement de cet algorithme avec les signaux faiblement corrélés, c'est à dire les tranches du signal de la parole non voisées, tandis que durant les zones voisées, nous avons remarqué un faible comportement de cet algorithme, et ce mauvais comportement se dégrade de plus en plus durant les tranches de silence du signal de la parole.

Cette manière aléatoire du comportement de l'algorithme NLMS avec le signal de la parole, nous à permis d'améliorer son comportement avec les différentes tranches (voisées, non voisées, silences), en faisant une adaptation adéquate du pas d'adaptation de l'algorithme en fonction de la nature des tranches du signal de la parole en cours d'utilisation, ceci a été réalisé comme suite:

1. Un faible pas d'adaptation durant les segments voisés pour affaiblir l'actualisation des coefficients du filtre adaptatif.
2. Un pas d'adaptation maximum de l'algorithme durant les tranches non voisée pour accélérer l'adaptation et par conséquent l'annulation d'écho acoustique.
3. Un blocage total de l'adaptation durant les zones de silences pour ne pas se trouver en situation de non-adaptation (blocage de l'algorithme), c'est à dire la dégradation de l'opération d'annulation d'écho acoustique.

IV.5.2. Performance du nouvel algorithme modifié NLMSmr

Nous avons réalisé trois applications avec l'algorithme NLMSmr. la première expérience correspond à l'identification de la réponse impulsionnelle d'un habitacle de voiture, la deuxième est celle d'une salle d'audioconférence et la troisième qui s'agit d'une salle de visioconférence, avec deux signaux de parole. On utilise dans ce cas l'algorithme NLMSmr (NLMS commandé par un système de segmentation récursivement).

- **Première application**

Pour cette application, la réponse impulsionnelle est celle d'un habitacle de voiture. Les résultats de simulations de notre algorithme adaptatif NLMS sont classés dans le tableau (IV.1).

Signal parole (2)						Signal parole (1)				
NLMSmr			NLMS			NLMSmr			NLMS	
L	μ_{nv}	μ_v	EQM	μ	EQM	μ_{nv}	μ_v	EQM	μ	EQM
32	1.0	0.27	-92.0	1.0	-90.4	1.0	0.65	-87.6	1.0	-86.6
64	1.0	0.27	-90.9	1.0	-89.4	1.0	0.65	-87.1	1.0	-85.7
128	1.0	0.27	-90.3	1.0	-85.8	1.0	0.65	-85.4	1.0	-84.2
256	1.0	0.27	-85.8	1.0	-76.6	1.0	0.65	-85.5	1.0	-83.1
512	1.0	0.27	-80.3	1.0	-78.2	1.0	0.65	-49.4	1.0	-46.4
1024	1.0	0.27	-56.4	1.0	-54.2	1.0	0.65	-40.5	1.0	-39.2

Tableau IV.1 : Résultats de l'EQM(dB) pour les différentes tailles du filtre L données, cas d'un habitacle de voiture.

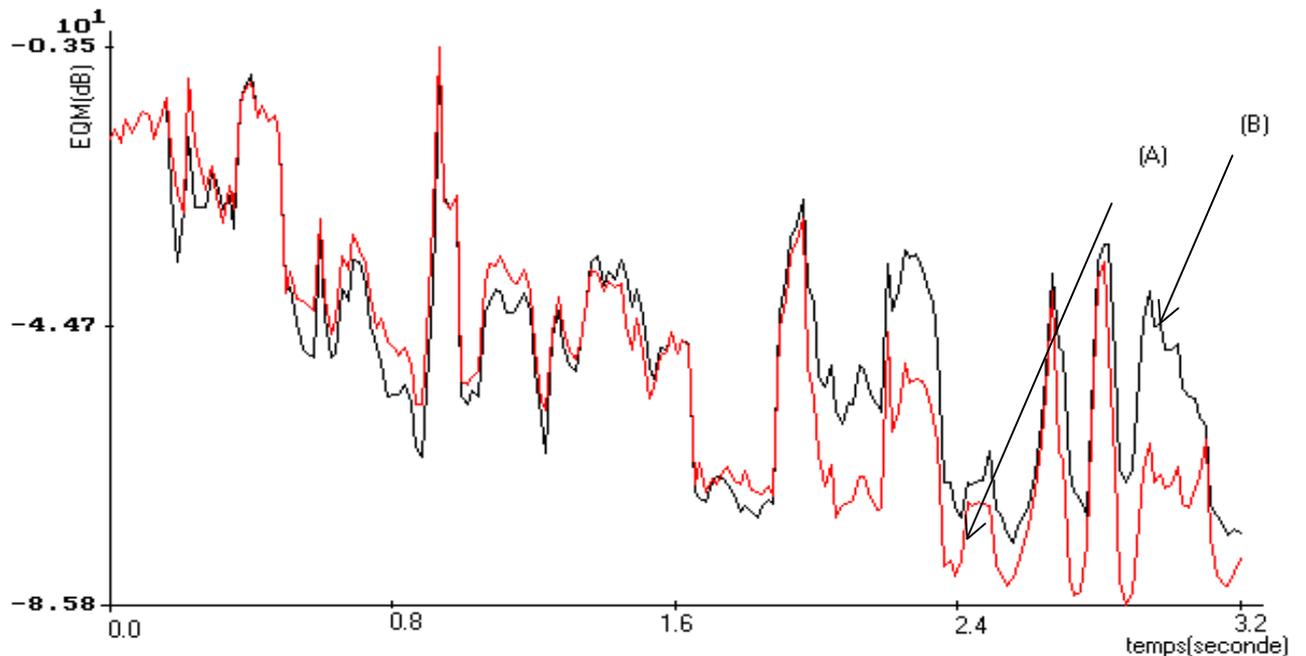


Figure IV.4 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS et NLMSmr, en utilisant la réponse impulsionnelle d'une voiture.

Taille du filtre $L = 256$, $\mu_{nv} = 1.0$, $\mu_v = 0.27$, signal de parole (2).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

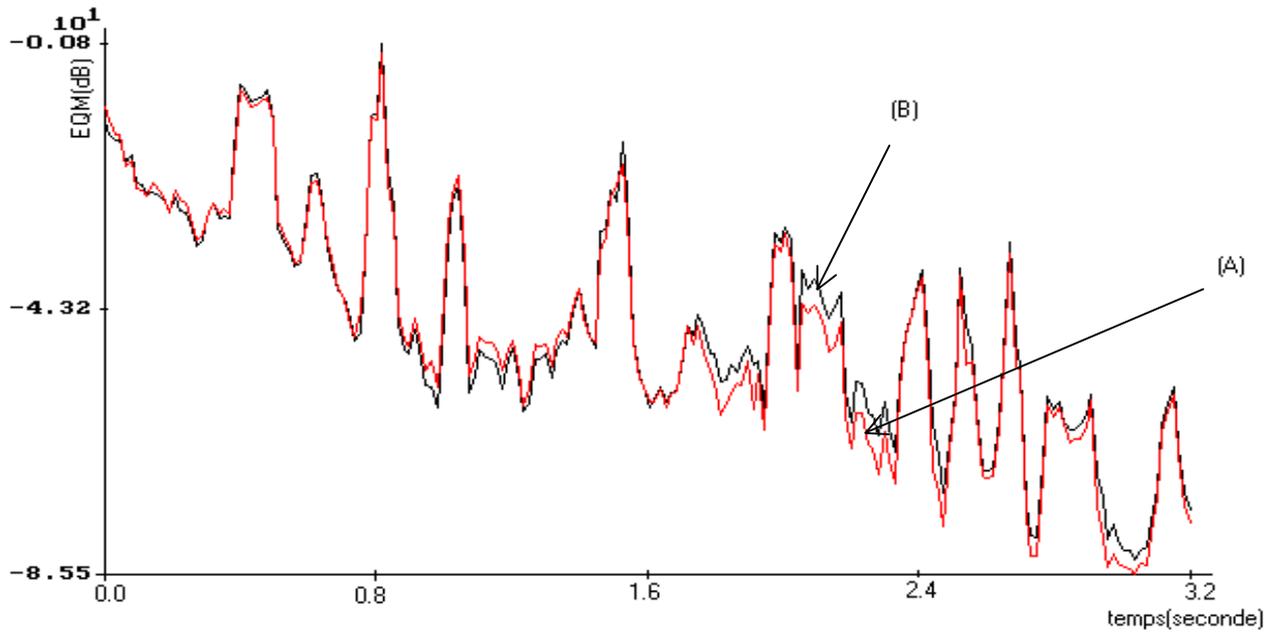


Figure IV.5 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMSmr, en utilisant la réponse impulsionnelle d'une voiture. Taille du filtre $L = 256$, $\mu_{nv} = 1.0$, $\mu\nu = 0.65$, signal de parole (1).
 (A) : algorithmme avec segmentation.
 (B) : algorithmme sans segmentation.

• **Deuxième application**

La réponse impulsionnelle dans ce cas est celle d'une salle d'audioconférence, (8192 points), les résultats de cette simulation sont portés sur le tableau IV.2.

Signal parole (2)					Signal parole (1)					
NLMSmr			NLMS		NLMSmr			NLMS		
L	μ_{nv}	$\mu\nu$	EQM	μ	EQM	μ_{nv}	$\mu\nu$	EQM	μ	EQM
32	0.9	0.26	-91.2	1.0	-89.5	0.9	0.26	-89.3	1.0	-86.7
64	0.9	0.26	-88.2	1.0	-85.7	0.9	0.26	-84.4	1.0	-83.7
128	0.9	0.26	-87.1	1.0	-83.1	0.9	0.26	-87.4	1.0	-82.4
256	0.9	0.26	-88.2	1.0	-79.5	0.9	0.26	-87.1	1.0	-83.5
512	0.9	0.26	-74.6	1.0	-74	0.9	0.26	-63.0	1.0	-57.8
1024	0.9	0.26	-52.0	1.0	-52.4	0.9	0.26	-39.5	1.0	-35.8
2048	0.9	0.26	-29.6	1.0	-31.5	0.9	0.26	-26.7	1.0	-27.8

Tableau IV.2 : Résultats de l'EQM(dB) pour les différentes tailles du filtre L données, cas d'une salle d'audioconférence.

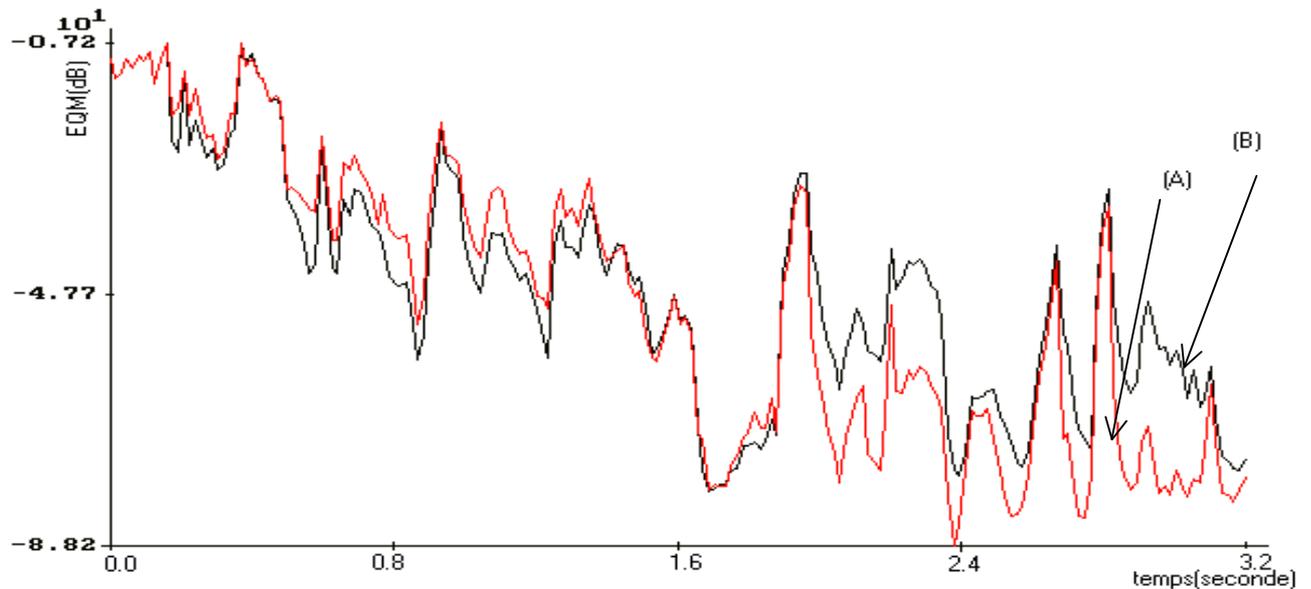


Figure IV.6 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMSmr, en utilisant la réponse impulsionnelle d'une salle d'audioconférence. Taille du filtre $L = 256$, $\mu_{nv} = 0.9$, $\mu_v = 0.26$, signal de parole (2).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

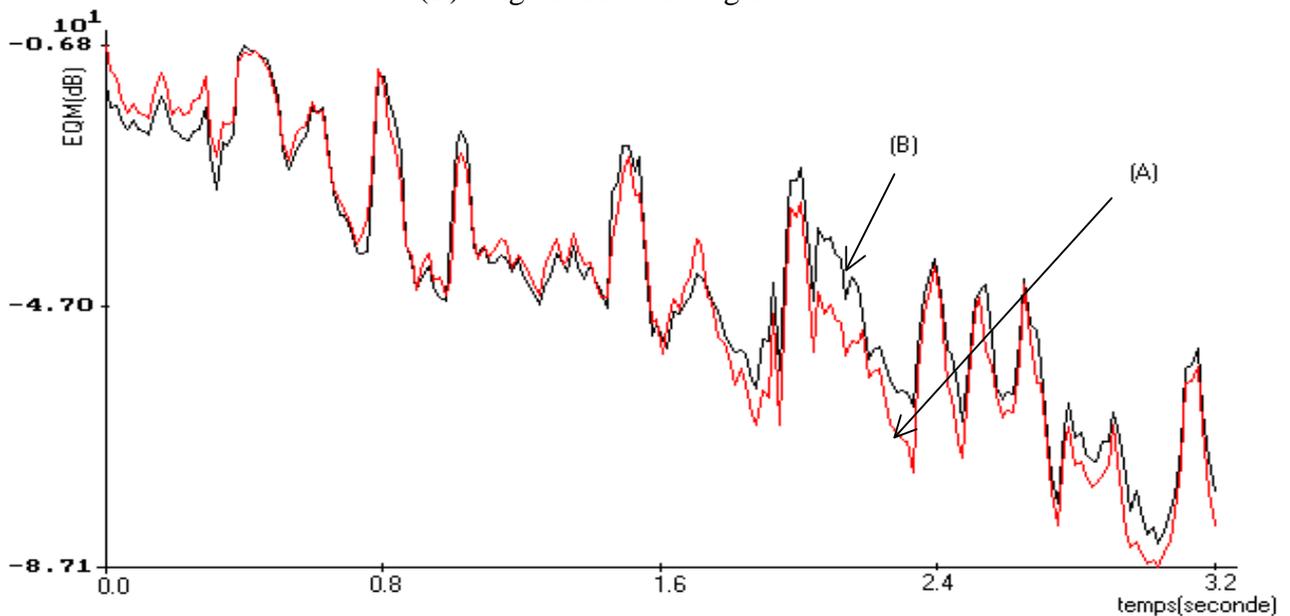


Figure IV.7 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMSmr, en utilisant la réponse impulsionnelle d'une salle d'audioconférence. Taille du filtre $L = 256$, $\mu_{nv} = 0.9$, $\mu_v = 0.26$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

• Troisième application

Dans cette application la réponse impulsionnelle est celle d'une salle de visioconférence. Les résultats des simulations sont illustrés dans le tableau IV.3.

Signal parole (2)						Signal parole (1)				
NLMSmr			NLMS			NLMSmr			NLMS	
L	μ_{nv}	μ_v	EQM	μ	EQM	μ_{nv}	μ_v	EQM	μ	EQM
32	1.0	0.3	-92.1	1.0	-90.4	1.0	0.3	-89.7	1.0	-86.6
64	1.0	0.3	-90.8	1.0	-89.4	1.0	0.3	-87.5	1.0	-85.7
128	1.0	0.3	-90.3	1.0	-85.8	1.0	0.3	-85.9	1.0	-84.2
256	1.0	0.3	-85.9	1.0	-76.6	1.0	0.3	-86.3	1.0	-83.2
512	1.0	0.3	-80.6	1.0	-78.2	1.0	0.3	-65.3	1.0	-62.2
1024	1.0	0.3	-54.4	1.0	-56.4	1.0	0.3	-48.7	1.0	-46.5
2048	1.0	0.3	-37.4	1.0	-40.7	1.0	0.3	-40.2	1.0	-39.2

Tableau IV.3 : Résultats de l'EQM(dB) pour les différentes tailles du filtre L données, cas d'une salle de visioconférence.

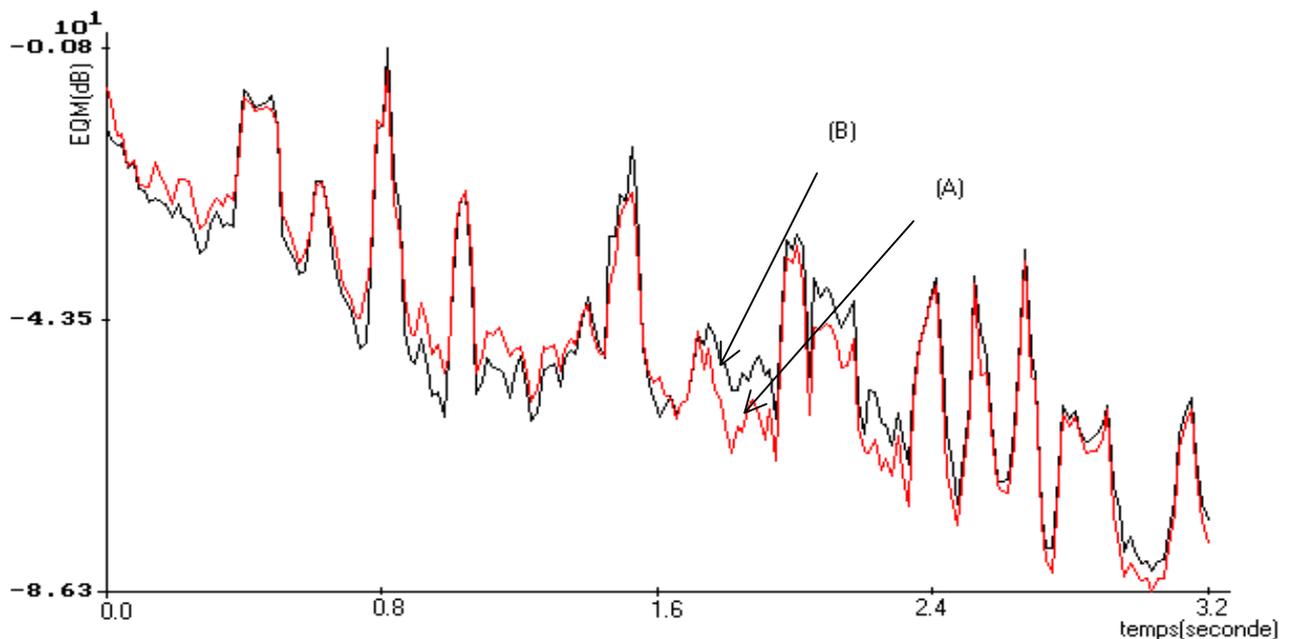


Figure IV.8 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMSmr, en utilisant la réponse impulsionnelle d'une salle de visioconférence. Taille du filtre $L = 256$, $\mu_{nv} = 1.0$, $\mu_v = 0.3$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

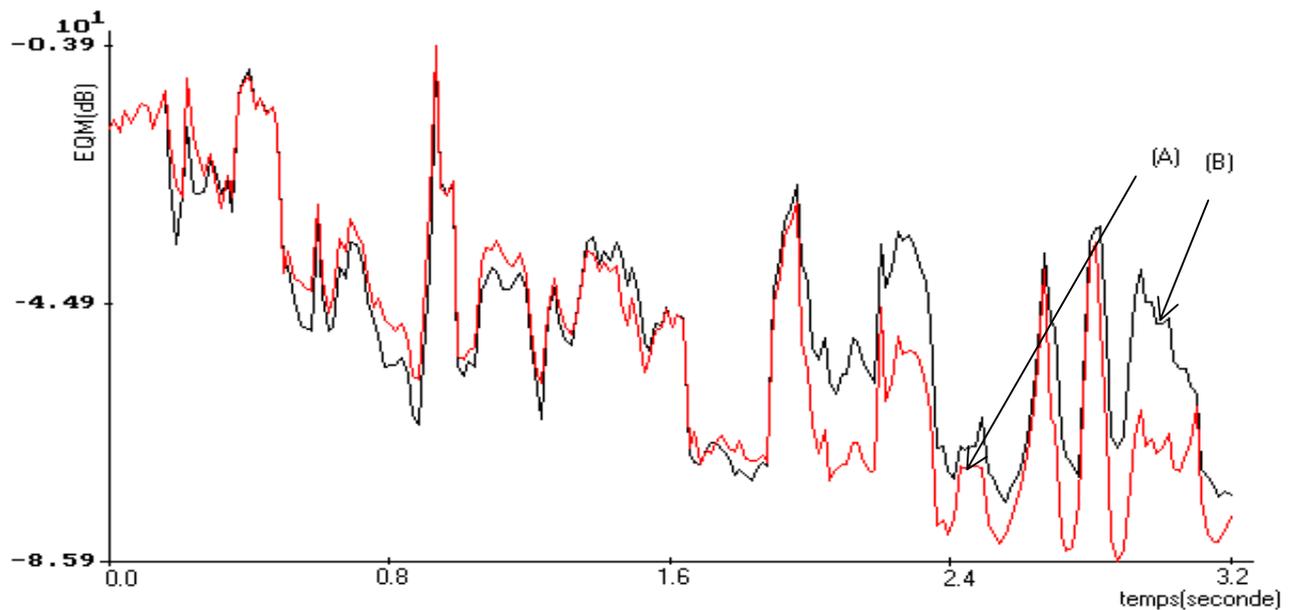


Figure IV.9 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMSmr, en utilisant la réponse impulsionnelle d'une salle de visioconférence. Taille du filtre $L = 256$, $\mu_{nv} = 1.0$, $\mu_v = 0.3$, signal de parole (2).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

- D'après les résultats de simulation classés dans les trois tableaux (IV.1, IV.2 et IV.3) nous remarquons que :

- Les erreurs quadratiques moyennes EQM(dB) exprimées en décibel de l'algorithme NLMS segmenté sont plus importantes que celles obtenues par l'algorithme NLMS non modifié, surtout pour les tailles inférieures à 512 points, avec une légère dégradation de ces derniers lorsque la taille du filtre transverse devient importante (exemple 1024, 2048 points avec le signal de parole 1).

- L'amélioration du fonctionnement de l'algorithme NLMS est due à l'adaptation du pas μ avec les différentes tranches du signal de la parole, de la, nous pouvons conclure la dépendance entre le comportement de l'algorithme NLMS et la nature du signal d'entrée.

- Les deux figures IV.4 et IV.5 présentent deux exemples de l'évolution temporelle de l'EQM de deux simulations caractéristiques en annulation d'écho acoustique. Nous représentons sur les deux figures l'évolution temporelle de l'EQM obtenue avec l'algorithme NLMS segmenté et celle de l'algorithme original, en utilisant une réponse impulsionnelle d'un habitacle de voiture. Nous notons que :

- Une certaine superposition apparaît au début des deux tracés des deux EQM, ce qui indique la présence de la zone non voisée, c'est à dire que les deux algorithmes ont le même pas d'adaptation et le même résultat de l'EQM.

- Une certaine amélioration de la version modifiée est remarquée au-dessous de -40dB de l'EQM et cette amélioration est obtenue grâce à l'augmentation du pas d'adaptation durant les zones non voisées, sa diminution durant les zones voisées et son blocage dans les zones de silence.

- Les deux figures IV.6 et IV.7 montrent bien l'amélioration du comportement de la version modifiée par rapport à celle non modifiée, dans l'affaiblissement de l'écho, ce dernier peut être atténué jusqu'à -88.2 dB pour la taille 256 points du filtre transverse (salle d'audioconférence, signal parole 2).

- Nous remarquons qu'au début des premières itérations de la Figure IV.8 le comportement de l'algorithme est aléatoire, c'est à dire que l'algorithme trouve beaucoup de difficultés pour retrouver ses coefficients nuls durant les zones de silence, ce qui mène à la dégradation de l'EQM, mais juste après 1.6 secondes, le fonctionnement de la version modifiée s'améliore et devient bon. Les deux figures IV.8 et IV.9 représentent l'évolution temporelle de l'EQM pour les deux versions modifiée et originale, mais cette fois ci, la réponse impulsionnelle est celle d'une salle de visioconférence.

- Nous notons toujours la dégradation de l'EQM durant les zones de silence car ces derniers ne permettent pas l'adaptation. En plus, on remarque que l'accélération de la vitesse de convergence qui dépend de la nature du signal excité.

IV.5.3. Résultat de simulation des algorithmes NLMS et NLMSmf

Nous avons appliqué la méthode de segmentation par des fenêtres fixes dans le domaine de l'annulation d'écho acoustique, par l'algorithme adaptatif du gradient stochastique normalisé NLMS. Nous avons réalisé deux applications avec le signal de la parole comme étant signal d'entrée. (Voir tableau IV.4 et IV.5).

• Première application

La réponse impulsionnelle est celle d'une salle d'audioconférence et les résultats de simulations de cette méthode avec l'algorithme NLMS sont présentés dans le tableau suivant :

Signal parole (2)						Signal parole (1)				
NLMSmf			NLMS			NLMSmf			NLMS	
L	μ_{nv}	μ_V	EQM	μ	EQM	μ_{nv}	μ_V	EQM	μ	EQM
32	0.85	0.7	-92.1	1.0	-91.0	0.85	0.3	-88.6	1.0	-87.0
128	0.85	0.7	-87.8	1.0	-82.3	0.85	0.3	-88.6	1.0	-81.5
256	0.85	0.7	-85.7	1.0	-79.1	0.85	0.3	-74.6	1.0	-63.6
512	0.85	0.7	-75.0	1.0	-74.8	0.85	0.3	-45.8	1.0	-44.8
1024	0.85	0.7	-52.9	1.0	-55.1	0.85	0.3	-30.2	1.0	-32.3
2048	0.85	0.7	-30.1	1.0	-30.9	0.85	0.3	-22.9	1.0	-25.8

Figure IV.4 : Résultats de l'EQM(dB) pour les différentes tailles du filtre L données, cas d'une salle d'audioconférence.

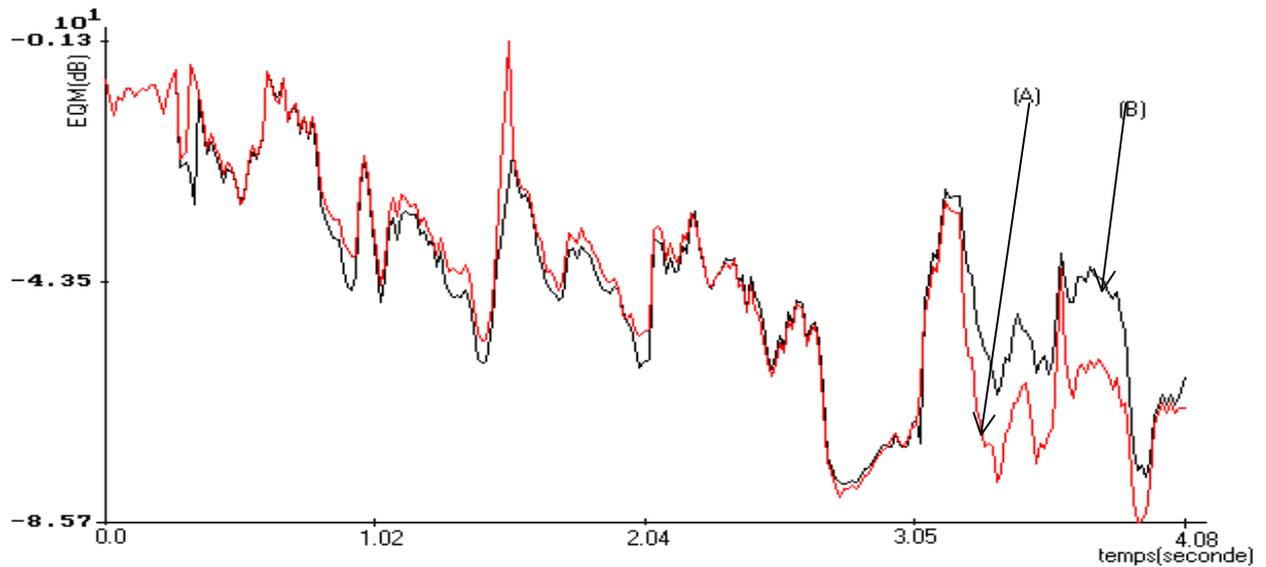


Figure IV.10 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMS avec segmentation, en utilisant la réponse impulsionnelle d'une salle d'audioconférence, segmentation par fenêtre fixe.

Taille du filtre $L = 256$, $\mu_{nv} = 0.85$, $\mu_v = 0.7$, signal de parole (2).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

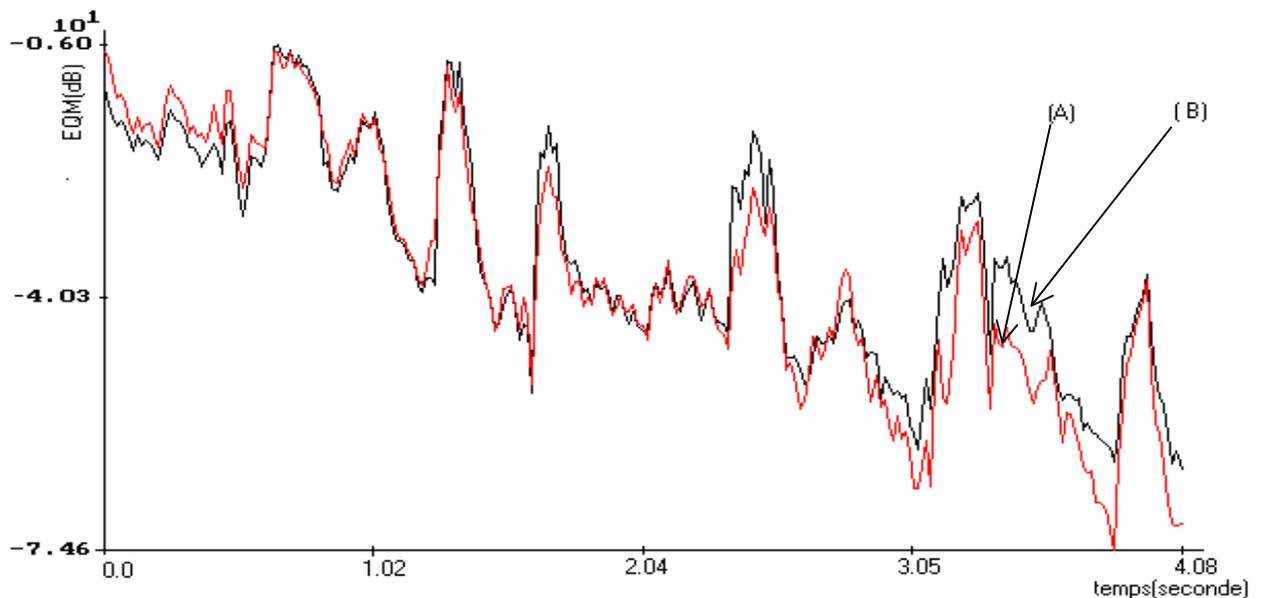


Figure IV.11 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMS avec segmentation, en utilisant la réponse impulsionnelle d'une salle d'audioconférence, segmentation par fenêtre fixe.

Taille du filtre $L = 256$, $\mu_{nv} = 0.85$, $\mu_v = 0.3$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

• Deuxième application

Pour cette application la réponse impulsionnelle est celle d'une salle de visioconférence (4096 points) et les résultats de simulations obtenus avec les deux signaux de la parole sont classés dans le tableau suivant :

Signal parole (2)						Signal parole (1)				
NLMSmf			NLMS			NLMSmf			NLMS	
L	μ_{nv}	μ_v	EQM	μ	EQM	μ_{nv}	μ_v	EQM	μ	EQM
32	0.85	0.7	-92.9	1.0	-91.6	0.85	0.3	-88.3	1.0	-86.6
128	0.85	0.7	-90.0	1.0	-85.9	0.85	0.3	-85.5	1.0	-82.7
256	0.85	0.7	-82.2	1.0	-73.5	0.85	0.3	-76.8	1.0	-69.2
512	0.85	0.7	-67.9	1.0	-69.2	0.85	0.3	-50.3	1.0	-49.0
1024	0.85	0.7	-52.9	1.0	-54.1	0.85	0.3	-39.9	1.0	-39.6
2048	0.85	0.7	-41.0	1.0	-42.8	0.85	0.3	-32.4	1.0	-34.4

Figure IV.5 : Résultats de l'EQM(dB) pour les différentes tailles de filtre L données, cas d'une salle de visioconférence.

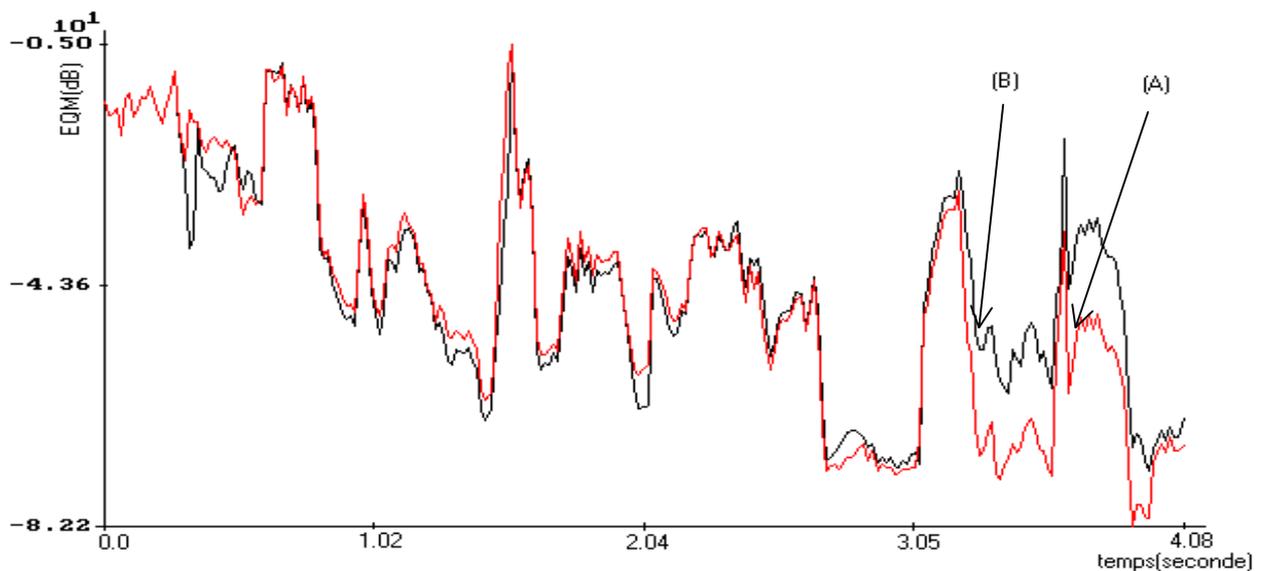


Figure IV.12 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMS avec segmentation, en utilisant la réponse impulsionnelle d'une salle de visioconférence, segmentation par fenêtre fixe.

Taille du filtre $L = 256$, $\mu_{nv} = 0.85$, $\mu_v = 0.7$, signal de parole (2).

(A) : algorithmme avec segmentation.

(B) : algorithmme sans segmentation.

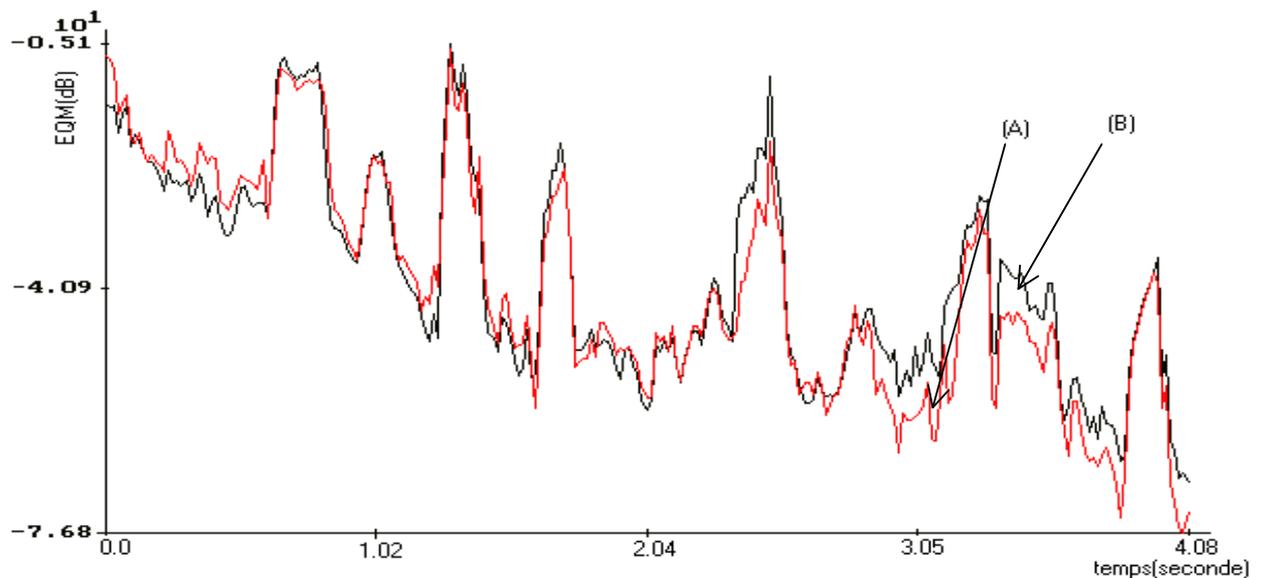


Figure IV.13 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : NLMS non segmenté et NLMS avec segmentation, en utilisant la réponse impulsionnelle d'une salle de visioconférence, segmentation par fenêtre fixe.

Taille du filtre $L = 256$, $\mu_{nv} = 0.85$, $\mu_v = 0.3$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

- Les deux tableaux IV.4 et IV.5, représentent les erreurs quadratiques moyennes EQM exprimées en décibel pour les deux versions de l'algorithme NLMS avec et sans segmentation et ceci pour différentes tailles du filtre adaptatif transverse L . D'après ces deux tableaux nous remarquons que :

- La version algorithmique modifiée permet d'avoir des erreurs quadratiques moyennes finales, plus importantes que celle de la version classique et ceci grâce à l'adaptation du pas μ en fonction des différentes tranches du signal de parole utilisé en test.

- Les tranches de silences perturbant le comportement de l'algorithme, car c'est un signal qui n'est pas informatif pour l'algorithme et par conséquent il dégrade son fonctionnement. La version modifiée permet de remédier à cet inconvénient en bloquant complètement l'adaptation des coefficients du filtre adaptatif dans les tranches de silence. Ceci a permis d'améliorer le fonctionnement de l'algorithme et son transitoire.

- Dès que nous dépassons 256 points, le régime permanent n'est jamais atteint et l'amélioration ne devient pas importante.

- Un comportement similaire est remarqué pour la version classique et modifiée par les deux signaux de parole avec une légère amélioration de l'EQM avec le signal de parole 2. Ceci montre bien que le transitoire avec ce type d'algorithme dépend de la nature du signal excitant.

- Les figures IV.10 et IV.11 représentent l'évolution temporelle de l'erreur quadratique moyenne EQM(dB) avec les deux signaux de la parole, la réponse impulsionnelle utilisée reste toujours celle d'une salle d'audioconférence.

- Nous remarquons un mauvais démarrage de la version modifiée par rapport à celle d'origine sur la Figure IV.11, ceci est logique dans le cas où nous savons très bien qu'au début de l'adaptation, les coefficients initiaux sont tous nuls (aléatoire) et si nous essayons d'appliquer notre technique, cette dernière ne permettra aucune amélioration si l'EQM temporelle est au dessous de -30dB , tandis qu'une amélioration nette est remarquée grâce à la technique de segmentation lorsqu'on est au dessus de -40dB de l'EQM, là où les coefficients du filtre adaptatif deviennent plus proches aux coefficients réels du système étudié.
- Nous notons aussi que la vitesse de convergence est bien améliorée entre 3.05 et 4.08 secondes, voir Figure (IV.10).
- Les deux Figures IV.12 et IV.13 montrent bien une nette amélioration du fonctionnement de l'algorithme modifié par rapport à celui non segmenté dans l'affaiblissement de l'écho qui peut être atténués jusqu'à -82.2 dB pour une taille 256 points du filtre transverse et ceci juste avant 4.08 secondes du temps d'opération de l'algorithme.
- Nous remarquons une de l'EQM dans les premières itérations de la figure (IV.13), ceci est dû au fait que l'algorithme n'est pas adapté.
- En fin, nous notons que la dépendance claire et nette entre la vitesse de convergence et les différentes tranches du signal excitant est toujours valable et est valide par les simulations.

IV.5.4. Etude des paramètres de l'algorithme FNTF VERSION 1

Pour faire améliorer le comportement de l'algorithme FNTF VERSION 1 dans le domaine d'annulation d'écho acoustique, nous avons réalisé plusieurs expériences sur les paramètres suivants :

E_0 : constante d'initialisation minimale des variances de prédiction aller / retour.

N : la taille des prédicteurs aller / retour.

λ : facteur d'oubli exponentiel.

L'évolution de l'effet de chaque paramètre cité au-dessus sur les performances de l'algorithme FNTF VERSION 1, et ceci pour plusieurs tailles du filtre transverse, ont conduis aux résultats suivants :

- **Effet de la constante d'initialisation E_0**

Tous les tests sur l'effet de la constante E_0 sur les performances de l'algorithme FNTF VERSION 1 ont montré que seulement le transitoire en phase initiale qui dépend de la constante E_0 . Ceci dit un bon choix de E_0 conduit à une bonne vitesse de convergence en phase initiale et assure un bon démarrage de l'algorithme. Alors qu'un mauvais choix diverge l'algorithme après une dizaine d'itération depuis son démarrage. En tout, on peut dire que l'effet de E_0 n'apparaît que dans la phase transitoire initiale.

- **Effet de la taille des prédicteurs aller/retour N**

Nous avons remarqué que la variation de ce paramètre important n'influe pas vraiment sur le comportement de l'algorithme si ce dernier est choisi supérieur à $2/3$ de la taille du filtre transverse L [1]. Les résultats de plusieurs simulations ont montré, également, que la modification de N après

le démarrage de l'algorithme n'a pas d'effet sur la vitesse de convergence et ceci est vrai à chaque fois que N s'approche de la taille du filtre L .

• **Effet du facteur d'oubli λ**

D'après les résultats des différents tests, qui ont été faits pour l'évolution de l'effet du facteur d'oubli exponentiel λ sur la performance de la vitesse de convergence de l'algorithme FNTF VERSION 1, nous avons remarqué que la performance du transitoire de cet algorithme est en fonction du choix du facteur λ , ce dernier est limité par la condition suivante :

$$\lambda > 1 - \frac{1}{2L+3.5} \quad (\text{IV.1})$$

Sachant que L représente la taille du filtre adaptatif.

Les tests ont montré qu'une bonne vitesse de convergence est obtenue en choisissant une valeur de λ proche de sa limite inférieure. Ce choix risque de rendre l'algorithme instable si on s'approche de cette limite. Tous les tests qui vont suivre sont basés sur l'adaptation du choix de λ en respectant toujours la condition (IV.1) en fonction des différentes tranches qui constituent le signal de parole utilisé en test.

IV.5.5. Modification de l'algorithme FNTF VERSION 1 (FNTF VERSION1mr)

D'après plusieurs simulations faites sur l'algorithme FNTF VERSION 1, nous avons constaté que, pour faire une adaptation adéquate du facteur d'oubli λ de l'algorithme FNTF VERSION 1 en fonction de la nature des segments du signal de la parole en cours d'utilisation, afin d'améliorer son comportement nous avons réagi comme suit :

1. Augmenter le facteur d'oubli λ durant les segments de silence par rapport à la limite de la condition de convergence (83).
2. Prendre un facteur d'oubli des segments voisés inférieur à celui des segments non voisés, ceci est valide toujours par rapport à la condition précédente (IV.183).
3. on a même pris le paramètres μ en pour commander l'adaptation et la non adaptation de cet algorithme en fonction de la nature du signal et elle a donnée les même résultats qu'en suivant les deux étapes 1 et 2.

Dans le reste des simulations nous avons retenu la méthode de commande par le facteur d'oubli exponentiel et les résultats sont présenté dans la suite de ce chapitre.

IV.5.6. Résultats de simulation de l'algorithme FNTF VERSION 1 et FNTF VERSION 1mr

Nous avons appliqué la méthode de segmentation récursive dans le domaine d'annulation d'écho acoustique par l'algorithme adaptatif FNTF VERSION 1 pour réaliser deux applications, la première est avec une réponse impulsionnelle d'une salle d'audioconférence et l'autre est avec une réponse d'une salle visioconférence. On note que les résultats d'application de la technique de segmentation par fenêtre a donnée pratiquement les mêmes résultats que la méthode récursive. Nous avons sélectionné la méthode de segmentation récursive pour montrer les résultats du nouveau algorithme FNTF avec segmentation (Algorithme FNTF VERSION 1mr).

- **Première application**

Pour cette application, la réponse impulsionnelle est celle d'une salle d'audioconférence, les résultats obtenus avec les deux signaux de parole sont illustrés sur les deux tableaux suivants :

Signal parole (2)							Signal parole (1)						
L	N	E_0	λ_v	λ_{nv}	λ_s	EQM	N	E_0	λ_v	λ_{nv}	λ_s	EQM	
256	200	2.0	0.9983	0.9985	0.9987	-93.7	256	2.0	0.9984	0.9985	0.9987	-92.1	
512	512	10.0	0.99968	0.99969	0.9997	-92.8	512	5.0	0.9995	0.99958	0.9996	-92.1	
1024	768	18.0	0.99956	0.99976	0.99986	-92.8	770	10.0	0.99979	0.9998	0.9999	-90.1	
2048	1600	12.0	0.99976	0.99982	0.99999	-64.0	1600	13.0	0.99977	0.99982	0.99999	-75.2	

Tableau IV.6 : EQM(dB) pour les L tailles de filtre données pour l'algorithme FNTF VERSION 1mr (cas d'une salle audioconférence).

Signal parole (2)					Signal parole (1)			
L	N	E_0	λ_s	EQM	N	E_0	λ_s	EQM
256	200	2.0	0.9987	-93.4	256	2.0	0.9987	-92.2
512	512	10.0	0.9997	-92.7	512	5.0	0.9996	-92.0
1024	768	18.0	0.99986	-84.0	770	10.0	0.9999	-78.0
2048	1600	12.0	0.99999	-60.3	1600	13.0	0.99999	-57.3

Tableau IV.7 : EQM(dB) pour les L tailles de filtre données pour l'algorithme FNTF VERSION 1 (cas d'une salle audioconférence).

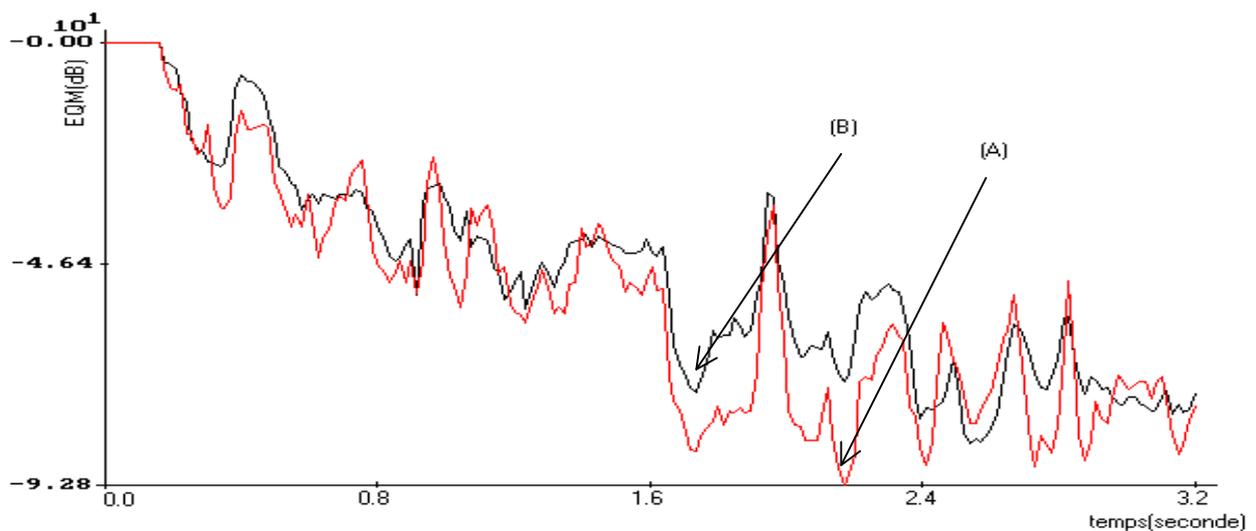


Figure IV.14: Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : FNTF VERSION 1 et FNTF VERSION 1mr, en utilisant la réponse impulsionnelle d'une salle d'audioconférence. Taille du filtre $L = 1024$, $\lambda_v=0.99956$, $\lambda_{nv}=0.99976$, $\lambda_s=0.99986$, $E_0=18$, $N=768$, signal de parole (2).
 (A) : algorithme avec segmentation.
 (B) : algorithme sans segmentation.

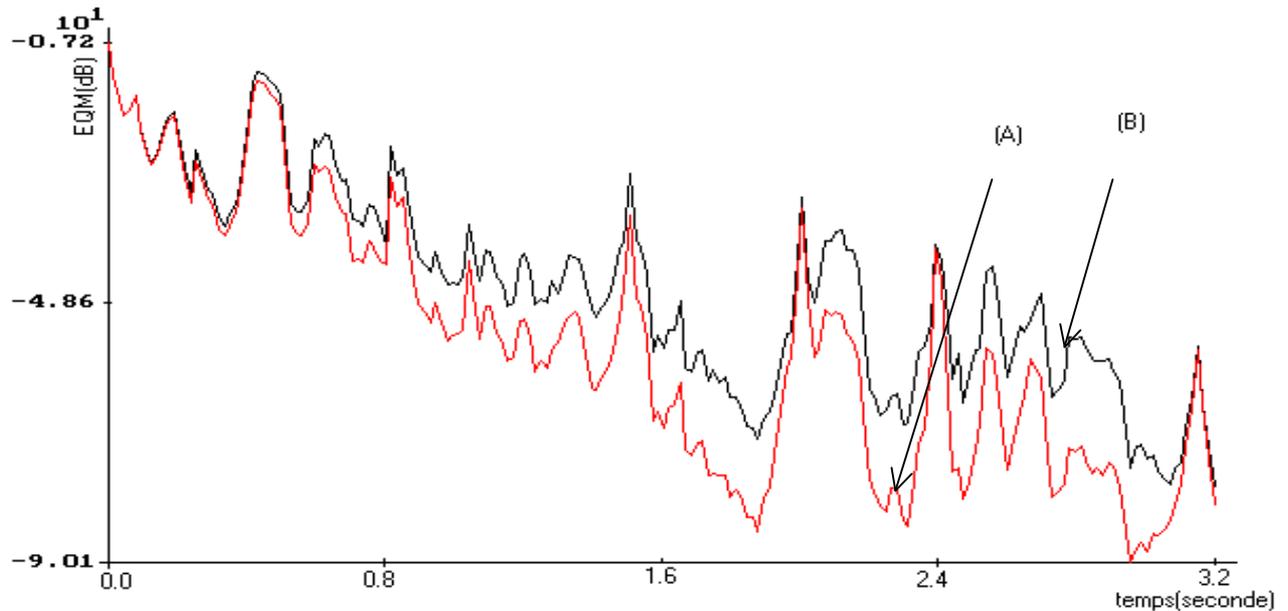


Figure IV.15: Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : FNTF VERSION 1 et FNTF VERSION 1mr , en utilisant la réponse impulsionnelle d'une salle d'audioconférence. Taille du filtre $L = 1024$, $\lambda_v = 0.99979$,

$\lambda_{nv} = 0.9998$, $\lambda_s = 0.9999$, $E_0 = 10$, $N = 770$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

• **Deuxième application**

La réponse impulsionnelle, dans ce cas, est celle d'une salle de visioconférence, les résultats de cette simulation sont portés sur les deux tableaux IV.8 et IV.9.

Signal parole (2)							Signal parole (1)					
L	N	E_0	λ_v	λ_{nv}	λ_s	EQM	N	E_0	λ_v	λ_{nv}	λ_s	EQM
256	200	5.5	0.9983	0.9985	0.9987	-90.6	256	2.0	0.9984	0.9985	0.9987	-92.4
512	512	10.0	0.9993	0.9995	0.999799	-92.0	512	10.0	0.9996	0.9997	0.9997	-93.7
1024	768	16.0	0.99956	0.99978	0.99986	-92.7	760	18.0	0.99958	0.9997	0.99989	-88.9
2048	1600	12.0	0.99977	0.99982	0.99999	-79.7	1600	13.0	0.9998	0.99987	0.99999	-82.7

Tableau IV.8 : EQM(dB) pour les L tailles de filtre données pour l'algorithme FNTF VERSION 1mr (cas d'une salle visioconférence).

Signal parole (2)					Signal parole (1)			
L	N	E_0	λ_s	EQM	N	E_0	λ_s	EQM
256	200	5.5	0.9987	-89.0	256	2.0	0.9987	-92.2
512	512	10.0	0.999799	-88.0	512	10.0	0.9997	-92.7
1024	768	16.0	0.99986	-86.8	760	18.0	0.99989	-85.9
2048	1600	12.0	0.99999	-65.0	1600	13.0	0.99997	-65.8

Tableau IV.9: EQM(dB) pour les L tailles de filtre données pour l'algorithme FNTF VERSION 1 (cas d'une salle visioconférence).

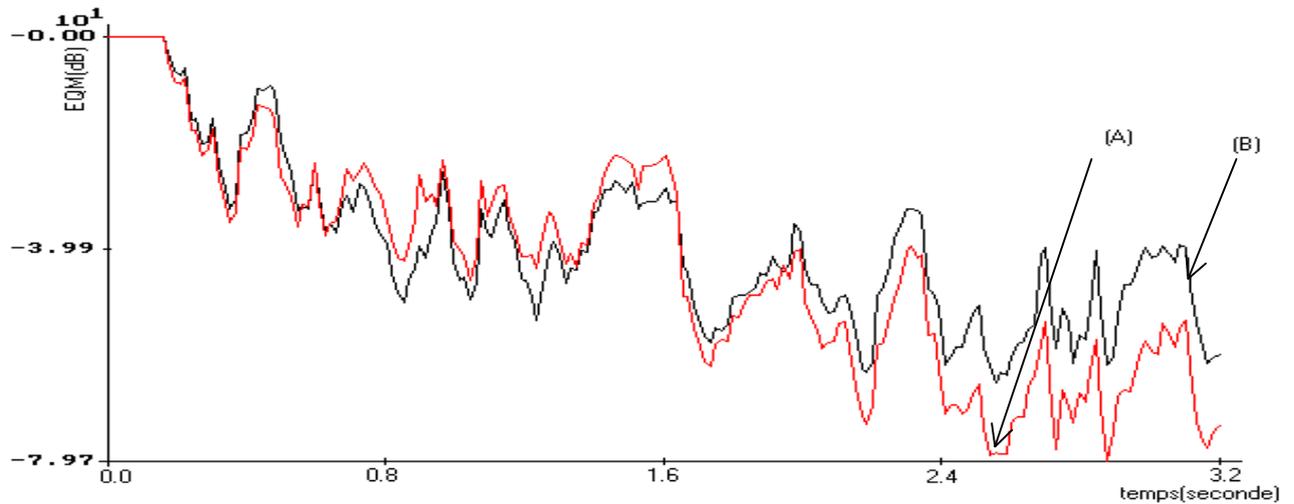


Figure IV.16 : Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : FNTF VERSION 1 et FNTF VERSION 1mr, en utilisant la réponse impulsionnelle d'une salle de visioconférence. Taille du filtre $L = 2048$,

$\lambda_v=0.99977$, $\lambda_{nv}=0.99982$, $\lambda_s=0.99999$, $E_0=12$, $N=1600$, signal de parole(2).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

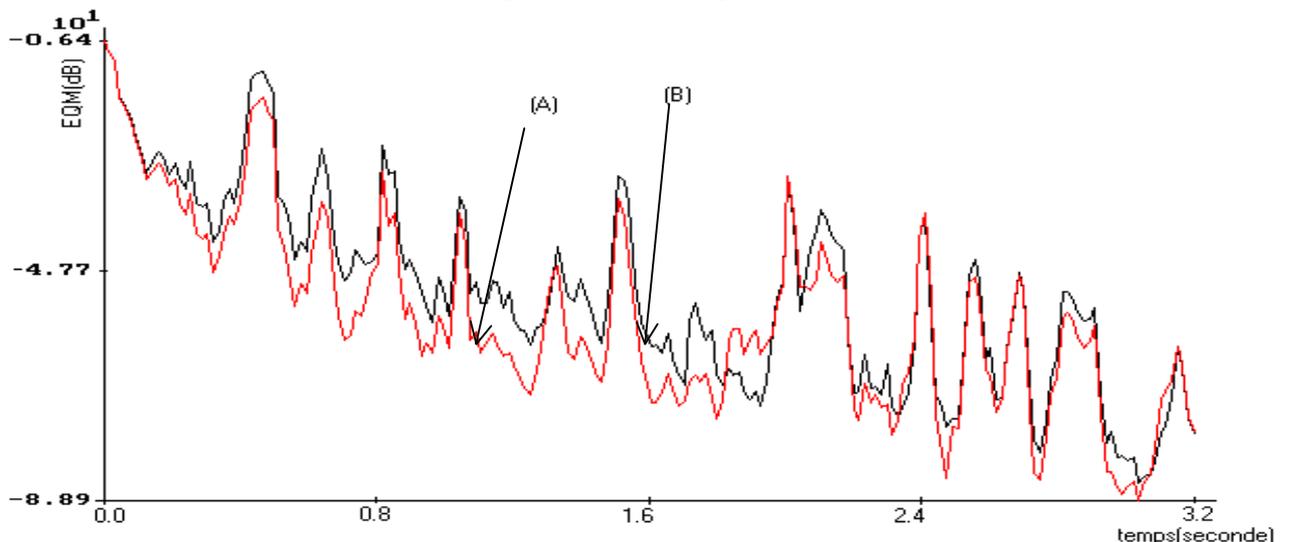


Figure IV.17: Evolution temporelle de l'EQM (énergie de l'erreur de filtrage) pour deux algorithmes : FNTF VERSION 1 et FNTF VERSION 1mr, en utilisant la réponse impulsionnelle d'une salle de visioconférence. Taille du filtre $L = 1024$, $\lambda_v=0.99958$, $\lambda_{nv}=0.9997$, $\lambda_s=0.99989$,

$E_0=18$, $N=760$, signal de parole (1).

(A) : algorithme avec segmentation.

(B) : algorithme sans segmentation.

- Les tableaux IV.6, IV.7, IV.8 et IV.9 montrent bien que les résultats des erreurs quadratiques moyennes de l'algorithme FNTF VERSION 1-MR, sont plus importantes que celle de la version classique. Ceci bien sur est dû à la technique d'adaptation du facteur d'oubli λ , donc ces résultats prouvent le bon comportement de cette version avec les différentes tranches du signal de la parole et ceci pour les tailles 256, 512, 1024 points avec une légère dégradation en ce qui concerne la taille 2048 points, cette dégradation devient moins importante surtout avec le signal de parole (1).

- Les figures IV.14 et IV.15 représentent l'évolution temporelle de l'EQM des deux algorithmes FNTF VERSION 1 avec et sans segmentation, pour deux signaux de parole, sachant que la réponse impulsionnelle dans ce cas est celle d'une salle d'audioconférence.

- Nous remarquons très bien l'amélioration qu'apportait la technique appliquée sur le comportement de la version modifiée, et ceci pour les tailles importantes comme la taille 1024, 2048 points, cette amélioration est possible si l'EQM est au-dessous de -5dB pour la figure IV.14 et -30dB pour celle IV.15. En plus la vitesse de convergence est grande, cela est très claire sur la figure IV.15.

- Les figures IV.16 et IV.17 montrent bien l'amélioration du fonctionnement de l'algorithme FNTF VERSION 1mr segmenté, cette amélioration est remarquée au dessous de -30dB.

- Nous remarquons, qu'au début du tracé IV.17, les deux signaux des deux algorithmes FNTF versions modifiée et celui du classique, sont superposés, cela signifie qu'ils ont le même facteur d'oubli exponentiel λ .

IV.6. Conclusion

En conclusion, nous pouvons dire que pour un signal d'entrée comme la parole, l'annulation d'écho acoustique avec les deux algorithmes (NLMS et FNTF VERSION 1) qui ont été présentés respectivement dans le chapitre II et III, et qui ont été pris comme modèle de deux grandes familles d'algorithmes (famille du gradient stochastique et celle des moindres carrés rapides), ont donné de très bons résultats en utilisant la technique proposée. Pour la segmentation du signal de parole, deux méthodes ont été utilisées, la première est la méthode récursive et la deuxième est la segmentation par des fenêtres fixes.

La technique que nous proposons est simple. Elle est basée d'une part sur l'intermittence du signal de parole et d'autre part sur la souplesse de l'algorithme pour être commandé via la nature des intermittence du signal de parole. Pour avoir une bonne amélioration sur le comportement de l'algorithme NLMS par exemple, nous avons fait une adaptation de son pas d'adaptation en fonction des différentes tranches du signal de la parole, ceci est réalisé comme suite :

- Un faible pas d'adaptation est choisi durant les segments voisés.
- Un pas d'adaptation maximum est choisi pour les segments non voisés.
- Un blocage total de l'adaptation pendant les périodes de silences.

Pour la famille des algorithmes de Newton, nous avons sélectionné la version 1 de l'algorithme FNTF proposée dans le chapitre III. Ceci est due à ces bonne performances et sa souplesse pour être mieux adapté à la technique proposée. Ceci est prouvé par la possibilité de sa commande en fonction du facteur d'oubli exponentiel et par le paramètre μ que nous avons introduit dans la partie de filtrage. Cette richesse en possibilité d'adaptation en fonction de l'intermittence du signal de parole, nous a permis de le sélectionner parmi les autres algorithmes pour le tester avec la technique proposée. On note aussi que cet algorithme a un grand avantage par rapport aux autres algorithmes, car il ne modifie pas la partie de prédiction mais seulement la partie de filtrage. L'adaptation de l'algorithme en question (FNTF VERSION 1mr) en fonction de la technique proposée se fait à travers le paramètre λ (facteur d'oubli exponentiel), suivant ces règles :

- Un facteur d'oubli exponentiel des zones voisées λ_v inférieur à celui des zones non voisées λ_{nv} .
- Un facteur d'oubli maximum pour les zones de silences λ_s .

Nous notons aussi que la seule contrainte rencontrée avec cet algorithme est la condition donnée par (IV.1 et III. 18) qui doit être obligatoirement vérifiée pour assurer un bon fonctionnement à long terme de l'algorithme de Newton qui est en cours d'utilisation.

Nous remarquons aussi que la plus grande vitesse de convergence a été prouvée par l'algorithme FNTF VERSION1mr en le comparant avec le NLMSmr/mf, cela est dû essentiellement à l'indépendance des performances de la famille des moindres carrés rapides et l'algorithme1 du FNTF VERSION1/mr, de la nature du signal d'entrée.

Nous concluons que la technique d'adaptation sélective que nous proposons dans ce chapitre permis une nette amélioration du fonctionnement des deux algorithmes dans les applications d'annulation d'écho acoustique, car les résultats des EQM finales sont très satisfaisantes pour les tailles inférieures à 512 points avec une dégradation de ces dernières pour les tailles importantes (1024, 2048 points), par conséquent, nous recommandons cette technique dans ce type d'application.

Conclusion générale et perspectives

Le travail que nous avons présenté dans cette thèse concerne l'utilisation pratique des algorithmes de filtrage adaptatif pour l'identification des filtres à réponses impulsionnelles acoustiques longues. Ces filtres apparaissent dans des applications d'annulation d'écho acoustique pour certains nouveaux terminaux des télécommunications, comme la téléconférence et le téléphone "mains libres".

Dans ce contexte, nous avons proposé cinq nouvelles versions de l'algorithme de filtrage adaptatif de type de Newton rapide (FNNTF). Le premier Algorithme proposé est basé sur une simple modification de la partie de filtrage par l'introduction d'un paramètre accélérateur. Le deuxième Algorithme est basé sur l'utilisation d'une technique de découpage temporelle dans l'actualisation des coefficients du filtre. Le troisième Algorithme est une modification du deuxième Algorithme, par l'utilisation de l'erreur de filtrage finale pour actualiser les coefficients du filtre. Le quatrième et cinquième Algorithmes sont basés, respectivement, sur la combinaison des propriétés du premier Algorithme avec les propriétés du deuxième et du troisième Algorithme.

Trois de ces algorithmes améliorent la vitesse de convergence dans le cas des systèmes invariants dans le temps (canaux acoustiques stationnaires). Les résultats des simulations ont également prouvé que tous les algorithmes proposés du FNNTF améliorent la capacité de poursuite quelque soient les signaux de test et les tailles des systèmes acoustiques à identifier, et une étude comparative détaillée entre ces algorithmes a été présentée[1].

Deux autres versions de l'algorithme FTF ont été présentées. Ces deux versions permettent un comportement similaire en performance de capacité de poursuite à celle de la version 4 et 5 de l'algorithme de Newton rapide (voir chapitre III). Ceci est due au fait que l'algorithme de Newton est une approximation stochastique de la matrice de covariance de l'algorithme FTF, donc ces deux algorithmes deviennent très proches en terme de performances lorsque la modélisation de la partie de prédiction du FNNTF devient proche de celle de la taille du filtre transverse des deux algorithmes FTF et du FNNTF.

Aussi une nouvelle version par bloc de l'algorithme FNNTF simplifié a été proposée et présentée. Cette version d'algorithme permet de réduire la complexité de calcul avec une certaine dégradation des performances de ce dernier. Sa bonne utilisation (exploitation) doit reposer sur certain choix optimum à faire entre la taille du bloc choisie et les performances souhaitées. Aussi, cette version ne permet pas une amélioration de la capacité de poursuite, ceci est due à la technique qui ne permet pas d'accéder aux paramètres de réglage de l'algorithme mais uniquement à la méthode de calcul des coefficients du filtre transverse.

Nous avons également observé que les propriétés de performance des nouveaux algorithmes FNTEF, (les cinq versions), dépendent directement du paramètre d'accélération μ et de la valeur du nombre de subdivisions P (même remarque pour les deux versions de l'algorithme FTF). Nous devons noter qu'aucun problème de divergence numérique n'a été expérimenté dans les simulations [47].

Pour la version par bloc de l'algorithme FNS, on remarque une dégradation plus importante si la taille du bloc d'actualisation est grande. Une bonne exploitation de cet algorithme nécessite un compromis entre ces performances en vitesse de convergence et en poursuite et la complexité de calcul souhaitée. Cet algorithme est important dans les applications en temps réel et en implémentation sur DSP [8,9,63].

Nous avons proposé dans le chapitre 4, une nouvelle technique qui permet d'améliorer l'annulation d'écho acoustique avec les algorithmes de filtrage adaptative transverse. Nous avons sélectionné deux types d'algorithmes pour montrer l'efficacité de cette technique dans ce domaine. La sélection de l'algorithme 1 dans la famille d'algorithmes des moindres carrés rapides (newton) est motivée par sa simplicité, sa robustesse aux accumulations des erreurs numériques et sa souplesse d'adaptation à la technique proposée. Les résultats obtenus avec cette technique sont très encourageants et montrent une nette amélioration des erreurs quadratiques moyennes par rapport à celles des versions originales.

Comme perspective à notre travail, nous proposons l'implémentation de tous ces algorithmes sur des DSP spécialisés tel que les TMS320C3x ou bien sur des stations VHDL.

Bibliographie

Chapitre I

- [1] A.Gilloire et J.P.Julien "L'acoustique des salles dans les télécommunications", L'écho des recherches, No 127. 1^{er} trimestre 1987.
- [2] J.P.Julien, A.Gilloire et A.Saliou, "Mesure de réponses impulsionnelles en acoustique", Note technique CENT NT/LAA/TSS181, Juillet 1984.
- [3] J.P.Julien, "Acoustique des salles, prise et restitution du son, analysées à partir de l' notion de canal acoustique", Note technique CENT NT/LAA/TSS/128, Août 1982.
- [4] P.Huyes and Al, "Adaptative filters - a review of techniques", BT Technology Journ. Vol.10 No 1 Jan. 1996.
- [5] A.Benveniste, M.Metivier et P.Priouret, "Algorithmes adaptatifs et approximations stochastiques :Théorie et Applications", dans Techniques stochastiques, MASSON 1987.
- [6] H.Schutze, Z.Ren, "Numerical characteristics of Fast Least Squares Transversal adaptation algorithms - A comparative study", Signal processing No 27, 1992.
- [7] A.Benallal, "Etude des algorithmes MCR et application à l'identification de réponses impulsionnelles acoustique", thèse de Doctorat, université de Rennes I, France Jan. 1989.
- [8] M.Djendi, " Réduction de le complexité des algorithmes des moindres carrées transversaux rapides, application à l'annulation d'écho acoustique et implantation sur la carte DSP TMS320C31", Thèse de Magistère, Université de BLIDA, Algerie, Juillet 2000.
- [9] M.Djendi, A.Benallal "Réduction de la Complexité des Calculs des Algorithmes d'Identification adaptative". Conférence Maghrébine en Génie Electrique CMGE'99, Université Mentouri, Constantine, Algérie 4-6 Décembre 1999.
- [10] G.Carayannis, D.Manolakis and N.Kalouptsidis, "A Unified View of Parametric Processing Algorithms for Prewindowed Signals", Signal Processing, vol-10, No 4, June 1986.
- [11] J.M.Cioffi and Th.Kailath, "Fast recursive least squares transversal filters for adaptative filtering", IEEE Trans. On Acoust. Speech and Signal Process. Vol-32, No 2, Apr.1984.
- [12] P.Petillon, A.Gilloire, S.Theodoridis : The Fast Newton Transversal Filter : An Efficient Scheme for Acoustic Echo Cancellation in Mobil Radio. IEEE Transaction on signal Processing, Vol.42, NO.3, March 1994
- [13] Panagiotis P. Mavridis and George V. Moustakidiss : Simplified Newton-Type Adaptive Estimation Algorithms. IEEE Transaction on signal processing, Vol.44, NO.8, August 1996.
- [14] M.G.Bellanger, "Engineering aspects of fast RLS algorithms in transversal adaptative filters", Proc. of IEEE ICASSP 1987, Dallas.

- [15] P.Fabre and C.Gueguen, "Fast RLS algorithms : Preventive divergence", Proc. of IEEE, ICASSP 1985, PP: 2350-2353, Tampa 1985, pp. 2350-2353
- [16] Z.J Mou, " Filtrage RIF rapide: algorithmes et architectures" Note technique CRPE/187 1990 (interne au CNET).
- [17] P. Duhamel, Z.J Mou, J. Benesty, " Une présentation unifiée du filtrage rapide fournissant tous les intermédiaires entre traitement temporel et fréquentiels", Proc. GRETSI 1989 pp 37-40.
- [18] J. Benesty, P. Duhamel, " A fast exact LMS adaptive algorithm", Proc. ICASSP 1990 pp. 1457-1460
- [19] J.S Soo, K.K Pang, " Multidelay block frequency domain adaptive filter", Proc of IEEE Trans. ASSP, vol 38, pp 373-376, Fev 1990.
- [20] M. Xu, Y. Grenier, " Time –frequency domain adaptive filters", Proc, ICASSP 1989, Glasgow, pp. 1154-1157.
- [21] O. Ait amrane, E. Moulines, Y. Grenier, " Identification de réponses impulsionnelles longues par gradient stochastique fréquentiel", Proc.GRETSI 1991, pp.481-484.
- [22] J.Benesty, L. Sheng Wen, P. Duhamel," Un algorithm du type LMS à capacité de poursuite et vitesse de convergence accrue", Proc.GRETSI 1991,pp.493-496.
- [23] J. Benesty," 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. Thèse de l'université de Paris-Sud.
- [24] M. Mboup, M. Bonnet," Une nouvelle structure blanchissante pour l'annulation d'écho acoustique", Proc.GRETSI, 1991,pp. 501-504.
- [25] A. Gilloire, M. Vetterli," Adaptive filtering in sub-bands with application to acoustic echo cancellers", IEEE. Trans. On signal Processing, Juillet/Aout 1992.
- [26] A.Gilloire, "Experiments with sub-bands acoustic echo cancellers for telconferencing", Proc. ICASSP 1987, pp. 2141-2144, Dallas.
- [27] M. Vetterli", A theory of multirate banks", IEEE. Trans on ASSP, vol. 35, pp. 356-359, March 1987.
- [28] M. Mboup, M. Bonnet, N. Bershad", Coupled adaptive prediction and systems identification. A statistical model and transient analysis", ICASSP 1992

Chapitre II

- [29] G.Moustakides, S.Theodoris :'' Fast Newton Transversal Filters – A New class of Adaptive Estimation Algorithms'', IEEE Trans. Signal Processing, vol. ASSP-39, n°10,PP.2184-2193,Oct.1991
- [30] S. Mac kay, " Modern spectral estimation", prentice-Hall Signal Processing Series 1988.
- [31] Van den bos," Alternative interpretation of Maximum Entropy Spectral Analysis,," IEEE trans, Information Theory pp.493-494, Jul 1971.
- [32] B. Picinbono, M. Baarret," Nouvelle présentation de la méthode du maximum d'entropie", Traitement du signal vol 7, pp. 153-1160, 1990.
- [33] P. Strobach," Least squares error modeling, Levinson & Shur algorithms", IEEE Signal Processing Magazine Jan 1991.
- [34] S. Theodoridis and G.V. Moustakides, "A novel class of fast adaptive algorithms for multichannel filtering",*Signal Processing V: Theories and Applications* (L. Torres et al., Eds.). Amsterdam: Elsevier, 1990, pp.409-412.
- [35] A. Benallal, " Étude des algorithmes MCR et application à l'identification de réponses impulsionnelles acoustiques", thèse de Doctorat, Université de Rennes I, France, Jan 1989.

- [36] A. Benallal and A. Gilloire, "A new method to stabilize fast RLS algorithms based on a first-order model of the propagation of numerical errors", *Proc. IEEE ICASSP 1988 Conf*, New York, April 1988, pp. 1373-1376.
- [37] P.Petillon, A.Gilloire, S.Theodoridis : The Fast Newton Transversal Filter : An Efficient Scheme for Acoustic Echo Cancellation in Mobil Radio. *IEEE Transaction on signal Processing*, Vol.42,NO.3, March 1994
- [38] M.Djendi "Réduction de la complexité de calculs des algorithmes des algorithmes adaptatifs, application à l'annulation d'écho acoustique et implantation sur la carte DSP TMS320C31, Thèse de Magistère, Université de BLIDA, Algérie juillet 2000.
- [39] Panagiotis P. Mavridis and George V. Moustakidiss, " Simplified Newton-Type Adaptive Estimation Algorithms", *IEEE Transaction on signal processing*, Vol.44, NO.8, August 1996.

Chapitre III

- [40] M. Djendi, M. Rahim, A. Guessoum, M. Bouchard, D. Berkani. "Comparative study of new version of the Newton type adaptive filtering algorithm", *Proc. of IEEE ICASSP 2004*, Montreal, Canada, May 2004, pp. 677-680.
- [41] M. Honigh and D.G Messerschmitt, *Adaptive Filters: Structure, Algorithms and Applications*. Boston: Kluwer, 1985.
- [42] M. Djendi, A. Benallal, A. Guessoum, D. Berkani, "Three new versions for the Newton type adaptive filtering algorithm", *Proc. of IEEE 7th International symposium on signal processing ISSPA'2003*, Paris, France, July 2003, vol.2, pp. 559-562.
- [43] A. Benallal and A. Gilloire, "Improvement of the convergence speed and of the tracking capability of the numerical Stable FRLS algorithm for adaptive filtering", *Proc. of IEEE ICASSP 1989 Conf.*, Glasgow, May 1989, pp. 1031-1034.
- [44] M. Djendi, A. Guessoum, A. Benallal and D. Berkani, "An accelerated Fast Newton Algorithm for the identification of time-varying systems with noisy outputs. Application to acoustic echo cancellation" *Proc. IEEE. 3rd ISSPIT*, Darmstadt, Germany, December 2003, pp. 644-647.
- [45] D.T.M Slock and T. Kailath, "Numerically stable fast RLS transversal filters", *Proc. IEEE ICASSP 1988 Conf*, New York, Apr. 1988, pp. 1365-1368.
- [46] M. Djendi, A. Guessoum, A. Benallal, M. Bouchard and D. Berkani, "Improvement of the tracking ability of the Newton type adaptive filtering algorithms", *Proc. IEEE ISCCSP 2004*, Hammamet, Tunisia, Mars 2004. pp. 701-704.
- [47] A.Benallal, A. Gilloire, "Instabilité et stabilité numérique des algorithmes de moindres carrés rapides excités par la parole", *XII ième Colloque GRETI*, Juan les Pins, juin 1989, pp.509-512.
- [48] B. Baykal and A.G.Constantinides, "Order-recursive undetermined recursive least-squares adaptive algorithms", *Signal Processing*, 1997, N°63(3), pp241-247
- [49] B. Baykal and A.G.Constantinides, "Undetermined-order recursive least-squares adaptive filtering, the concept and algorithms", *IEEE Transaction on Signal Processing*, 1997, 45(2), pp.346-362.
- [50] k.Ozeki and T. Umeda, " An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties", *Electron.Commun.Japan*, 1984, 67-A(5), pp.19-27.
- [51] S.Gay and S.Tavathia, "The fast affine projection algorithm", *Proc. IEEE ICASSP 1995*, 1995, vol.5, pp.3023-3026.

- [52] E. Eleftheriou, D.Falconer, "Tracking properties and steady-state performances of RLS adaptive algorithms", IEEE Trans. ASSP, 1986, Vol.34 no.5, pp.1097-1109.
- [53] O. Macchi, " A general methodology for comparison of adaptive filtering algorithms in a nonstationary context", in Signal Processing V: Theories and Applications (L. Torres et al. Eds), Amsterdam: Elsevier, 1990, pp.189-192.
- [54] M. Djendi, M. Bouchard, A. Guessoum, A. Benallal, D. Berkani, "Improvement of the Convergence Speed and the Tracking Ability of the Fast Newton Type Adaptive Filtering (FNTF) Algorithm", accepted, to appear in Signal Processing Journal.
- [55] M. Djendi, A. Benallal, A. Guessoum, D. Berkani, "New combinatorial method for the improvement of the convergence speed and the tracking abilities of the Fast Stable RLS algorithm", Proceeding of the 37th IEEE.Asilomar conference on signals, systems and computers, CALIFORNIA, USA, November 2003.
- [56] M. Djendi, A. Guessoum, A. Benallal, M. Bouchard, " New Combinatorial Methods for the Improvement of the Convergence Speed and the Tracking Ability of the Fast Stable RLS Adaptive Algorithm", Proceeding of IEEE First International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2004, Hammamet, Tunisia, Mars 2004. pp. 681-684.
- [57] M. Rezigui et K. Otsmane, " Identification des systèmes acoustiques en temps réel sur la carte DSP AU32", Thèse d'ingénieur d'état, université de Blida 2000.
- [58] M. Bachouti et F. Moussi, "Etude de la capacité de poursuite des algorithmes des moindres carrés transversaux rapides. Application aux canaux de couplage acoustique mesurés par la carte OROS-AU32 ", Thèse d'ingénieur d'état, Institut d'électronique, Université de Blida, Octobre 2001.
- [59] A. Hadj Idris et S. Dekik, "Amélioration du comportement des algorithmes des moindres carrés rapides de type Newton et application à l'annulation d'écho acoustique", Thèse d'ingénieur d'état, Institut d'électronique, Université de Blida, Octobre 2001..

Chapitre VI

- [60] D. Boucherf, M. Djendi, A. Guessoum, C. Ait Kaci, " Improvement of Acoustic Echo Cancellation by Gradient and Newton Adaptive Filtering Algorithms", **accepted**, to appear in Proceeding of IEEE Second International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2006, Morocco, 2006.
- [61] C. Ait Kaci, D. Boucherf, "Amélioration du comportement des algorithmes de filtrage adaptatif transverse avec le signal de parole", PFE d'ingénieur d'état, Institut d'Electronique, Université de Blida 2002.
- [62] M. Lebgaa, M. Khelich, "Segmentation automatique de la parole", PFE d'ingénieur d'état, Institut d'Electronique Université de Blida. Octobre 2000.

Conclusion générale

- [63] M. Djendi, A.Benallal, and A.Guessoum,"Implementation of the Adaptive Identification algorithms with a Reduced Complexity on the card OROS-AU32 (TMS320C31)", National conference on automatism and signals, SNAS' 02, Annaba Algeria, October 2002.

Communications de Mr. Mohamed Djendi

- M. Djendi, Master thesis: "Complexity reduction of transversal adaptive filtering algorithm of Newton and implementation on DSP OROS TMS320C31", Blida University, Algeria, July 2000 (in French).
- M. Djendi, Engineer thesis: "study and realization of acoustic impulse responses measurement system" Blida University - Algeria 1997 (in French)
- M. Djendi, DEUA thesis: "study and realization of a remote control of garage door", Blida University, Algeria, June 1994 (in French).
- M. Djendi, M. Rahim, A. Guessoum, M. Bouchard and D. Berkani, "Comparative study of new versions of the Newton type adaptive filtering algorithm", Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2004, vol. 2, pp.677-680, Montreal, May 2004
- M. Djendi, A. Guessoum, M. Bouchard, A. Benallal, D. Berkani, "Improvement of the Convergence Speed and the Tracking Ability of the Fast Newton Type Adaptive Filtering (FNTF) Algorithm", accepted, to appear in Signal Processing.
- M. Djendi, A. Gilloire, P. Scalart, Noise cancellation using two closely spaced microphones: experimental study with a specific model and two adaptive algorithms. Accepted to appear in Proceeding of IEEE, ICASSP 2006.
- D. Boucherf, M. Djendi, A. Guessoum, C. Ait Kaci, "Improvement of Acoustic Echo Cancellation by Gradient and Newton Adaptive Filtering Algorithms", accepted, to appear in Proceeding of IEEE Second International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2006, Morocco, 2006.
- A. Benallal, A. Benkrid, M. Djendi, S. Kerfi, "Measurement and identification of acoustic impulse responses", Conference ICCCP'05, Koweit, 2005
- M. Djendi, A. Guessoum, A. Benallal, M. Bouchard, "New Combinatorial Methods for the Improvement of the Convergence Speed and the Tracking Ability of the Fast Stable RLS Adaptive Algorithm", Proceeding of IEEE First International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2004, Hammamet, Tunisia, Mars 2004. pp. 681-684.
- M. Djendi, A. Guessoum, A. Benallal, M. Bouchard and D. Berkani, "Improvement of the tracking ability of the Newton type adaptive filtering algorithms", Proceeding of IEEE First International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2004, Hammamet, Tunisia, Mars 2004. pp. 701-704.
- M. Djendi, A. Guessoum, A. Benallal, D. Berkani, "An Accelerated Fast Newton algorithm for the identification of time-varying systems with noisy outputs. Application to acoustic echo cancellation", Proceeding of IEEE. ISSPIT.2003, International symposium on signal processing and information technology", DARMSTADT, ALLEMAGNE December 2003.
- M. Djendi, A. Benallal, A. Guessoum, D. Berkani, "New combinatorial method for the improvement of the convergence speed and the tracking abilities of the Fast Stable RLS algorithm", Proceeding of the 37th IEEE. Asilomar conference on signals, systems and computers, CALIFORNIA, USA, November 2003.
- M. Djendi, A. Benallal, A. Guessoum, and D. Berkani, "Three new versions for the Newton adaptive filtering algorithm", Seventh IEEE International Symposium on Signal Processing and its Applications IEEE. ISSPA 2003 1-4 July 2003, Paris, FRANCE, Vol2, PP.559-562.
- M. Djendi, A. Benallal, and A. Guessoum, "Implementation of the Adaptive Identification algorithms with a Reduced Complexity on the card OROS-AU32

- (TMS320C31)", National conference on the automatism and the signals, SNAS' 02, Annaba Algeria, October 2002.
- M. Djendi, A. Benallal, and A. Guessoum, " Acoustic echo cancellation using adaptive newton algorithm", National conference on signals and automatics, SNAS'02", Annaba Algeria, Octobre 2002. (in French).
 - M. Djendi, A Benallal , M. Mekrazi, S. Kerefi, "Realisation of a measurement system of acoustics impulse responses", International conference SSA'2 99 (10-12 May 1999 – Blida – Algeria). (in French)
 - M. Djendi, A Benallal . Fast Adaptive algorithms with reduced complexity, North African Conference CMGE'99 (4-6 Dec. 1999 – Constantine – Algeria), (in French)
 - M. Djendi, A. Benallal, and A. Guessoum, " Acoustic echo cancellation using adaptive newton algorithm", National conference on signals and automatics, SNAS'02", Annaba Algeria, Octobre 2002. (in French).

Biographie de Mr Mohamed DJENDI

Mohamed DJENDI

Mohamed DJENDI

School address

Ecole ENSSAT - Université de RENNES 1,
 INRIA/IRISA - Equipe de recherche R2D2,
 BP 80518 - 6 rue de Kerampont,
 Lannion 23000,
 France,
 Office tel.: +33 (0)2 96 46 9157
 Fax: +33 (0)2 96 46 9075
 Mobile tel.: +33 (0)6 73 32 94 62
 E-mails: mohamed.djendi@enssat.fr, m_djendi@yahoo.fr



EDUCATION

PHDs

1- PhD In Signal processing (in preparation since March 2004)
 The Engineering School ENSSAT at the University of Rennes 1, Lannion, France.
Topic : " Study of the sound recording in a compact terminal inside a car ”
Advisors : Professors: Pascal Scalart and André Gilloire

2- PhD In Signal and Communications (Achieved in 19 September 2006)
 Electrical Engineering Department, Ecole Nationale Polytechnique, 10 Avenue Hassan Badi,
 B.P. 182 El-Harrach, Algiers 16000, Algeria;
Topic : " Study of fast Newton transversal adaptive filtering algorithms for the identification of
 acoustics impulse responses".
Advisors : Professors: Abderrezak. Guessoum and Daoud Berkani

Master

In Signal processing and telecommunications (1997/98 to 1999/2000)
 University Saad Dahalab of Blida, Algeria.
Topic : Complexity reduction of transversal adaptive filtering algorithm and implementation on
 DSP of type OROS TMS320C31, (Blida University, jully 2000)
Advisor: Dr. Ahmed Benallal.

Engineer of state

In electronics engineering (communication) (1994/95 to 1996/97)
 University Saad Dahalab of Blida, Algeria.
Topic: study and realization of acoustic impulse responses measurement system (Blida
 university , september 1997)
Advisor: Dr. Ahmed Benallal

DEUA of state

In control engineering (1990/91 to 1993/94)
Topic: study and realization of a remote control of garage door (Blida, University, 1994)
Advisor: Dr. Ahmed Benallal.

PROFESSIONAL EXPERIENCE

- Teaching**
- Assistant Professor at the university of Blida – Electronics Institute (1998-2001).
 - Assistant Professor at the university of USTHB (Algiers) – Electronics Institute (2002-2003).
 - Associate Professor at the university of Blida – Electronics Institute (2001-2005).

Taught various undergraduate courses such as:

- Theory and practice of signal processing.
- Digital Signal Processing.
- Main functions of electronics (amplification, modulation, commutation techniques)
- Identification and modelling of signals and systems.
- Analysis, synthesis and design of digital filters.
- Adaptive identification and adaptive digital filters.
- Instrumentation
- Micro-wave (hyper frequency)
- Computer science (programming)

Director of studies Electronics institute. University of Blida (2003-2004)

- Research**
- Full member in LATSI research laboratory, Blida University (2000-2005)
 - Fellow researcher in LATSI research laboratory, Blida University (2003-2005)

Field of interests: Adaptive identification, Signal processing, Speech processing, acoustics, DSP implementation, Blind source separation, noise and echo cancellation, Adaptive filtering.

Projects:

- Member of team – Acoustics in Room – Institute of electronics (2000-2003).
- Member of team – Advanced systems for signal and data – Institute of electronics, (03-05)
- Member of team – Research on Speech Coding and DSP implementation – institute of electronics (03-05).
- Member of team – Arabic characters recognition – electronics institute (03-05).

Advisor Advised various projects for undergraduate and graduate students in electronics and signal processing. (see the last page for more details about these projects)

- 07 projects of Electronic DEUA (2000-2005),
- 06 projects of electronics engineers (2000-2005),
- 02 projects of post-graduate masters (2000, 2005).

OTHERS

Languages Arabic, English and French.

Computer skills Windows 2000/XP professional, Microsoft office 2000/XP, Pascal compiler, Matlab, computer hardware, C, TMS320C31 programming language.

Interests Reading, sport (football and Athletics) and drawing.

PUBLICATIONS, COMMUNICATIONS

- **M. Djendi**, Master thesis: “Complexity reduction of transversal adaptive filtering algorithm of Newton and implementation on DSP OROS TMS320C31”, Blida University, Algeria, July 2000 (in French).
- **M. Djendi**, Engineer thesis: “study and realization of acoustic impulse responses measurement system” Blida University - Algeria 1997 (in French)

- **M. Djendi**, DEUA thesis: “study and realization of a remote control of garage door”, Blida University, Algeria, June 1994 (in French).
- **M. Djendi, A. Gilloire, P. Scalart**, “ Frequency Domain post-filters for noise cancellation using two closely spaced microphones”, *Submitted to ICASSP 2007, Hawaii, USA*.
- **M. Djendi, P. Scalart, A. Gilloire**, “ noise cancellation by blind source separation structure”, Project of patent in noise and echo cancellation with University of Rennes-1 (ENSSAT) and the Group of France-Telecom (to appear in 2006).
- **M. Djendi, A. Guessoum, M. Bouchard, A. Benallal, D. Berkani**, "Improvement of the Convergence Speed and the Tracking Ability of the Fast Newton Type Adaptive Filtering (FNTF) Algorithm", *Journal of Signal Processing*, vol.86, n.7, pp.1704-1719, July 2006.
- **M. Djendi, A. Gilloire, P. Scalart**, Noise cancellation using two closely spaced microphones: experimental study with a specific model and two adaptive algorithms. *Proceeding of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2006*, vol.3, PP.774-747, Toulouse, France, May 2006.
- **D. Boucherf, M. Djendi, A. Guessoum, C. Ait Kaci**, " Improvement of Acoustic Echo Cancellation by Gradient and Newton Adaptive Filtering Algorithms", *In (CDROM) of IEEE Second International Symposium in Communication, Control and Signal Processing, IEEE ISCCSP2006*, Morocco, 2006.
- **A. Benallal, A. Benkrif, M. Djendi, S. Kerfi**, "Measurement and identification of acoustic impulse responses", *Conference ICCCP'05*, Koweit, 2005
- **M. Djendi, M. Rahim, A. Guessoum, M. Bouchard and D. Berkani**, "Comparative study of new versions of the Newton type adaptive filtering algorithm", *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2004*, vol. 2, pp.677-680, Montreal, May 2004
- **M. Djendi, A. Guessoum, A. Benallal, M. Bouchard**, " New Combinatorial Methods for the Improvement of the Convergence Speed and the Tracking Ability of the Fast Stable RLS Adaptive Algorithm", *Proceeding of IEEE First International Symposium in Communication, Control and Signal Processing, IEEE. ISCCSP 2004*, Hammamet, Tunisia, Mars 2004. pp. 681-684.
- **M. Djendi, A. Guessoum, A. Benallal, M. Bouchard and D. Berkani**, "Improvement of the tracking ability of the Newton type adaptive filtering algorithms", *Proceeding of IEEE First International Symposium in Communication, Control and Signal Processing, IEEE.ISCCSP 2004*, Hammamet, Tunisia, Mars 2004. pp. 701-704.
- **M. Djendi, A. Guessoum, A. Benallal, D. Berkani**, "An Accelerated Fast Newton algorithm for the identification of time-varying systems with noisy outputs. Application to acoustic echo cancellation", *Proceeding of IEEE. ISSPIT.2003*, International symposium on signal processing and information technology", DARMSTADT, ALLEMAGNE December 2003.
- **M. Djendi, A. Benallal, A. Guessoum, D. Berkani**, "New combinatorial method for the improvement of the convergence speed and the tracking abilities of the Fast Stable RLS algorithm", *Proceeding of the 37th IEEE. Asilomar conference on signals, systems and computers, CALIFORNIA, USA, November 2003*.
- **M. Djendi, A. Benallal, A. Guessoum, and D. Berkani**, " Three new versions for the Newton adaptive filtering algorithm", *Seventh IEEE International Symposium on Signal Processing and its Applications IEEE.ISSPA 2003 1-4 July 2003, Paris, FRANCE, Vol2, PP.559-562*.
- **M. Djendi, A. Benallal, and A. Guessoum**, "Implementation of the Adaptive Identification algorithms with a Reduced Complexity on the card OROS-AU32 (TMS320C31)", *National conference on the automatism and the signals, SNAS' 02*, Annaba Algeria, October 2002.
- **M. Djendi, A. Benallal, and A. Guessoum**, " Acoustic echo cancellation using adaptive newton algorithm", *National conference on signals and automatics, SNAS'02*, Annaba Algeria, Octobre 2002. (in French).
- **M. Djendi, A. Benallal, M. Mekrazi, S. Kerefi**, "Realisation of a measurement system of acoustics impulse responses", *International conference SSA'2 99 (10-12 May 1999 – Blida – Algeria)*. (in French)
- **M. Djendi, A. Benallal** . Fast Adaptive algorithms with reduced complexity, *North African Conference CMGE'99 (4-6 Dec. 1999 – Constantine – Algeria)*, (in French)
- **M. Djendi, A. Benallal, and A. Guessoum**, " Acoustic echo cancellation using adaptive newton algorithm", *National conference on signals and automatics, SNAS'02*, Annaba Algeria, Octobre 2002. (in French).

References

Dr. Ahmed Benallal

University of Demam, Arabie Saoudite, Arabie Saoudite

Pr. Abderrezak Guessoum	Email: a_benallal@hotmail.com University Saad Dahlab of Blida, Algeria
André Gilloire	Email: aguessoum@hotmail.com France Telecom Research Centre (CNET/DIH/LAN). Route de Trégastel - Lannion - France. Email: andre.gilloire@cnet.francetelecom.fr
Martin Bouchard	University of Ottawa, Email: bouchard@site.uottawa.ca
Pr. Pascal Scalart	University of Rennes - IRISA / ENSSAT, 6 Rue de Kerampont, Lannion, France. Email: scalart.pascal@enssat.fr

**Projects of DEUA (DEUA: Diploma of higher applied education),
Advisor: Mohamed Djendi**

- "Study and realization of an infra-red remote control ". Project of DEUA. Institute of Electronics University of Blida. June 1999.
- "Study and realization of a voice faker ". Project of DEUA. Institut d' Electronique University of Blida. June 1999.
- "Study and realization of a control device containing microcontrolor assisted by a PC (Personnal Computer), Project of DEUA". Institute of Electronics University of Blida. July 2003.
- "Study and realization of a control device per telephone". Project of DEUA. Institute of Electronics University of Blida. July 2003
- "Study and realization of a system protection against the cuts of the sector". Project of DEUA. Institute of Electronics University of Blida. July 2004
- "Study and Realization of an infra-red remote control based on a PIC(Microcontroler) ", PFE DEUA. Institute of Electronics University of Blida. July 2005.
- "Study and realization of an control device per phone card", Poject of DEUA. Institute of Electronics University of Blida. July 2005

**Projects of Ingeener state
Advisor: Mohamed Djendi**

- "Identification of system in real time on DSP AU32 (TMS320C31)", Project of Engineer state. Electronics Institute of University of Blida, October 2000.
- "automatic speech segmentation", Project of Engineer state, Institute of Electronics University of Blida, October 2000.
- "Improvement of the behavior of FTF and Newton algorithm. Application to the acoustic Echo cancellation", Project of Engineer state. Electronics Institute, University of Blida, October 2001.
- "Study of the tracking capabilty of the FTF algorithms and application to the acoustic impulse response meauserd by that card DSP AU32", Project of Engineer state. Electronics Institute, University of Blida, June 2001.
- "Improvement of the tracability of Newton adaptive algorithm", Project of Engineer state, Electronics Institute, University of Blida, June 2002.
- "Improvement of the FTF Adaptive filtering algorithm with speech as input", Project of Engeener state, Electronics Institute, Unievrsty of Blida, June 2002.

**Projects of Master state
Advisor: Mohamed Djendi with Prof Abderrezak Guessoum**

- "Improvement of FTF and Newton adaptive filtering algorithms ", Master thesis, Aeronautics Institute, Blida University, Algeria, September 2003.
- "acoustic Cancellation of echo by Newton adaptive filtering algorithms ", Master thesis, being prepared (proposed in September 2004), Electronic Institute, University of Blida, It will be achieved in July 2006, Algeria.

Certificates of honor

- 1994: (DEUA), Major of promotion, (Project mention: very honourable),
- 1997: (Engineer), Major of promotion, (Project mention: very honourable),
- 2000: (Master), Project mention, very honourable.
- 2006: (PHD), Project mention, very honourable.