

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur 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 ÉLECTRONIQUE

Présentée par :

AREZKI Madjid

ALGORITHMES ADAPTATIFS ÉVOLUÉS
DE FILTRAGE

Membres du jury

Président : M. F. BOUDJEMA, Professeur, ENP Alger
Examineurs : M. N.K. M'SIRDI, Professeur, LSIS Marseille (France)
M. A. OULD ALI, Professeur, EMP Alger
M. A. OUAHABI, Professeur, Université Tours (France)
Mme. A. SERIR, Maître de Conférences, USTHB
Directeur de thèse : M. D. BERKANI, Professeur, ENP Alger
Co-directeur de thèse : M. A. GUESSOUM, Professeur, USD Blida.

ENP 2011

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur 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 ÉLECTRONIQUE

Présentée par :

AREZKI Madjid

ALGORITHMES ADAPTATIFS ÉVOLUÉS DE FILTRAGE

Membres du jury

Président : M. F. BOUDJEMA, Professeur, ENP Alger
Examineurs : M. N.K. M'SIRDI, Professeur, LSIS Marseille (France)
M. A. OULD ALI, Professeur, EMP Alger
M. A. OUAHABI, Professeur, Université Tours (France)
Mme. A. SERIR, Maître de Conférences, USTHB
Directeur de thèse : M. D. BERKANI, Professeur, ENP Alger
Co-directeur de thèse : M. A. GUESSOUM, Professeur, USD Blida.

ENP 11

ملخص

الهدف الرئيسي من دراستنا هو تقييم إمكانية تطوير خوارزميات التكيف المتعلقة بتطبيقات نظام إلغاء الصدى السمعي مع المحافظة على التوازن بين تعقد عمليات حسابه المختصرة و قدراته التكيفية. نعرض أساليب جديدة لخوارزميات التريبيعات المصغرة الممكن تكرارها السريعة و المستقرة رقمياً (NS-FRLS). يستعمل للحصول عليها بعض الصيغ المطبقة المتوفرة في خوارزميات التريبيعات المصغرة الممكن تكرارها السريعة (FRLS) من أجل تقدير الأخطاء الرقمية و استدراكها عند أي نقطة من الخوارزم و ذلك لتغيير خواصها الرقمية. تشكل هذه الخوارزميات حملاً كبيراً من العمليات الحسابية التي يجب البحث عن اختصارها. و لهذا نقترح خوارزم سريع جديد (M-SMFTF) من أجل الترشيح التكيفي. يقدم هذا الخوارزم فائدة أكيدة في تكيف المرشحات الطويلة جداً مثل التي تستعمل في مسائل إلغاء الصدى السمعي، بالنظر إلى تعقده المنخفض، استقراره الرقمي و تقاربه مع وجود إشارة الصوت. يساوي تعقد عملياته الحسابية $6L$ و يختصر هذا التعقد الحسابي بطريقة مثيرة إلى $(2L+4P)$ عندما نستعمل متنبئ رواح مختصر رتبته P ($P \ll L$)

الكلمات المفتاح: RLS السريع، LMS مسوى، ترشيح تكيفي، سرعة التقارب، قدرة الإلتباع، إشارة الصوت

Résumé

Le principal objectif de notre étude, est d'évaluer la possibilité de développer des algorithmes d'adaptation pour les applications de système d'annulation d'écho acoustique tout en maintenant un équilibre entre sa complexité de calcul réduite et ses performances adaptatives. Nous présentons de nouvelles versions d'algorithmes des moindres carrés récursifs rapides numériquement stables (NS-FRLS). Elles sont obtenues en utilisant quelques formules redondantes, disponible dans les algorithmes des moindres carrés récursifs rapides (FRLS), pour estimer les erreurs numériques et rétroagir ces dernières en un point quelconque de l'algorithme afin de modifier ses propriétés numériques. Ces algorithmes représentent une charge de calcul trop importante qu'il faut chercher à réduire. Nous proposons ainsi un nouvel algorithme rapide M-SMFTF pour le filtrage adaptatif. Cet algorithme présente un intérêt certain, pour l'adaptation de filtres très longs comme ceux utilisés dans les problèmes d'annulation d'écho acoustique, à cause de sa complexité réduite, sa stabilité numérique et sa convergence en présence du signal de parole. Sa complexité de calcul est de $6L$ et celle-ci est considérablement réduite à $(2L+4P)$ quand on utilise un prédicteur aller réduit d'ordre P ($P \ll L$).

Mot-clef: RLS rapide, LMS normalisé, Filtrage adaptatif, Vitesse de convergence, Capacité de poursuite, Signal de parole.

Abstract

The main objective of our study is to evaluate the possibility to develop algorithms of adaptation for the applications system of acoustic echo cancellation, while maintaining equilibrium between its reduced calculation complexity and its adaptive performances. We present new algorithms versions of fast recursive least squares numerically stable (NS-FRLS). They are obtained by means of redundant formulas, available in the fast recursive least squares (FRLS) algorithms, to estimate numerical errors and to retroact them in an unspecified point of the algorithm in order to modify its numerical properties. These algorithms represent a very important load of calculation that needs to be reduced. We propose thus a new fast algorithm M-SMFTF for adaptive filtering. This algorithm presents a certain interest, for the adaptation of very long filters, like those used in the problems of echo acoustic cancellation, due to its reduced complexity, its numerical stability and its convergence in the presence of the speech signal. Its calculation complexity is of $6L$ and this is considerably reduced to $(2L+4P)$ when we use a reduced P -size ($P \ll L$) forward predictor.

Keyword: Fast RLS, NLMS, Adaptive Filtering, Convergence Speed, Tracking capability, Speech signal.

Remerciements

Le travail présenté dans cette thèse a été réalisé au Laboratoire Signal et Communications du département d'Electronique de l'Ecole Nationale Polytechnique (ENP) d'El-Harrach et au Laboratoire de Traitement de Signal et Image (LATSI) du département d'Electronique de l'Université de Blida.

Je tiens à remercier sincèrement Monsieur Farès BOUDJEMA, Professeur à l'Ecole Nationale Polytechnique (ENP) d'El-Harrach, qui m'a fait l'honneur d'accepter la présidence du jury.

Je remercie vivement Monsieur Nacer-Kouider M'SIRDI, Professeur à l'Université de Marseille (France), Monsieur Abdelaziz OULD ALI, Professeur à l'Ecole Militaire Polytechnique (EMP) d'Alger, Monsieur Abdeldjalil OUAHABI, Professeur à l'Université de Tours (France) et Madame Amina SERIR, Maître de Conférences à l'Université Houari Boumediene (USTHB), qui m'ont fait l'honneur de participer au jury.

Je tiens à présenter mes vifs remerciements à mon Directeur de thèse Monsieur Daoud BERKANI, Professeur à l'Ecole Nationale Polytechnique (ENP) d'El-Harrach, et mon Co-Directeur Monsieur Abderezzak GUESSOUM, Professeur à l'Université de Blida.

Mes plus sincères remerciements sont adressés à Monsieur Ahmed BENALLAL, ancien Maître de Conférences à l'Université de Blida, pour ses conseils tout au long de la thèse avec qui j'ai eu de fructueuses discussions théoriques et techniques.

Mes remerciements s'adressent aussi à mon frère Yazid pour son aide très précieuse en anglais.

Je voudrais terminer ces remerciements par les personnes les plus importantes: Ma famille et surtout mes parents qui m'ont toujours soutenu.

A la mémoire de mon père

Table de matières

Principales notations et abréviations	1
Liste des tableaux et figures	2
INTRODUCTION GENERALE	6
Chapitre I. IDENTIFICATION PAR ALGORITHMES ADAPTATIFS.	10
I.1 Introduction	10
I.2 Identification par algorithmes adaptatifs	13
I.3 Présentation des algorithmes	16
I.3.1 Famille du gradient stochastique (LMS)	17
I.3.2 Famille d'Algorithmes des moindres carrés récursifs (RLS)	19
I.3.2.1 Algorithme des moindres carrés récursifs (RLS)	19
I.3.2.2 Algorithmes des moindres carrés récursifs rapides (FRLS)	21
I.4 Conclusion	28
Chapitre II. ANALYSE DES ERREURS NUMERIQUES DES ALGORITHMES FRLS	29
II.1 Introduction	29
II.2 Divergence des algorithmes FRLS	29
II.3 Stabilité numérique des algorithmes FRLS	30
II.3.1 Approche et Hypothèses	30
II.3.2 Nouvelle méthode de stabilisation	32
II.3.3 Nouvelles versions d'algorithmes FRLS	34
II.3.4 Résultats de simulation	36
II.4 Propagation des erreurs numériques de la partie prédiction	40
II.4.1 Calcul des équations de propagation des erreurs dans les variables aller	40
II.4.2 Calcul des équations de propagation des erreurs dans les variables de Kalman	41
II.4.3 Calcul des équations de propagation des erreurs dans les variables retour	48
II.4.4 Résumé	50
II.5 Conclusion	52
Chapitre III. STABILITE NUMERIQUE DES ALGORITHMES FRLS	53
III.1 Introduction.	53
III.2 Analyse des erreurs numériques de la partie prédiction.	53
III.2.1 Analyse des erreurs numériques	53
III.2.2 Choix des paramètres de contrôle et condition de stabilité	56
III.3 Analyse de Convergence	62
III.3.1 Convergence en moyenne	63
III.3.2 Convergence en quadratique moyenne	63
III.4 Résultats de simulations	65

III.4.1 Comportement des algorithmes FRLS avec des signaux stationnaire	65
III.4.1.1 Partie prédiction	65
III.4.1.2 Partie filtrage	72
III.4.2 Comportement des algorithmes FRLS avec des signaux parfaitement prédictibles. . .	75
III.4.2.1 Cas d'une sinusoïde pure	76
III.4.2.2 Cas de plusieurs sinusoïdes pures	80
III.4.2.3 Influence d'un bruit additif	81
III.4.3 Comportement des algorithmes FRLS avec des signaux non stationnaire.	82
III.5 Conclusion	85
Chapitre IV. ALGORITHME M-SMFTF	86
IV.1 Introduction	86
IV.2 Algorithmes adaptatifs	87
IV.3 Réduction de Complexité	90
IV.3.1 Proposition de l'algorithme M-SMFTF	90
IV.3.2 Proposition de l'algorithme RM-SMFTF	94
IV.4 Analyse des erreurs de propagation.	96
IV.4.1 Analyse des erreurs numériques de la partie prédiction	96
IV.4.2 Analyse de convergence.	102
IV.5 Résultats de simulations	103
IV.5.1 Comparaison des performances des algorithmes SMFTF et M-SMFTF.	103
IV.5.2 Comportement des algorithmes M-SMFTF et RM-SMFTF.	105
IV.5.2.1 Influence des valeurs η et c_a sur l'algorithme M-SMFTF	105
IV.5.2.2 Influence de la valeur η sur l'algorithme RM-SMFTF	105
IV.5.3 Comparaison des performances du M-SMFTF avec le NS-FRLS et le NLMS.	109
IV.5.4 Effet de l'ordre du prédicteur P sur l'algorithme RM-SMFTF	113
IV.6 Conclusion	114
Chapitre V. APPLICATION A L'ANNULATION D'ECHO ACOUSTIQUE	115
V.1 Introduction.	115
V.2 Principe de l'annulation d'écho acoustique.	115
V.3 Descriptions des systèmes et des signaux utilisés	118
V.4 Comparaison des performances différents algorithmes.	120
V.5 Résultats de simulations	121
V.5.1 Canal stationnaire avec entrée stationnaire.	121
V.5.2 Canal non stationnaire avec entrée stationnaire.	122
V.5.3 Canal stationnaire avec entrée non stationnaire	132
V.6 Conclusion.	136
CONCLUSION GENERALE	137
Bibliographie.	140

Principales notations et abréviations

• Conventions

Une variable scalaire sera notée en italique et en petite lettre « x »

Une variable vectorielle sera notée en gras et en petite lettre « \mathbf{x} »

Une variable matricielle sera notée en gras et en majuscule « \mathbf{X} »

Une constante sera notée en italique et majuscule « C »

$(.)^T$ Opérateur de transposition

$E\{.\}$ Espérance mathématique

$\text{tr} [.]$ Trace d'une matrice

$\|.\|$ Norme euclidienne d'un vecteur

$\log_{10}(.)$ Logarithme à base 10

• Abréviations

AR(M): AutoRégressif d'ordre M

FIR: Finite Impulse Response (réponse impulsionnelle finie)

LMS: Least Mean Square (gradient stochastique)

NLMS: Normalized Least Mean Square (gradient stochastique normalisé)

RLS : Recursive Least Squares (moindres carrés récursif)

FRLS : Fast Recursive Least Squares (moindres carrés récursif rapide)

NS-FRLS: Numerically Stable Fast Recursive Least Squares (moindres carrés récursif rapide numériquement stable)

FAEST: Fast A-posteriori Error Sequential Technique

FTF: Fast Transversal Filter (filtre transverse rapide)

FK: Fast Kalman (Kalman rapide)

KaGE: Kalman Gain Estimator (estimateur de gain de Kalman)

FNTF: Fast Newton Transversal Filter (filtre transverse de Newton rapide)

FBPLS: Fast Backward Predictor Least Squares (prédiction retour des moindres carrés rapides)

BRLS: Block RLS (RLS par bloc)

SU RLS: Subsampled Updating RLS

FSU SFTF: Fast SU Stable FTF

FSU FNTF: Fast SU FNFT

SMFTF: Simplified type FTF (FTF simplifié)

M-SMFTF: Modified and Simplified FTF (FTF simplifié et modifié)

RM-SMFTF: Reduced size predictor M-SMFTF (M-SMFTF à prédicteur réduit)

$SNR(n)$: Signal Noise to ratio (rapport signal à bruit)

$MSE(n)$: Mean Square error (erreur quadratique moyenne)

$ERLE(n)$: Echo Return Loss Enhancement (amélioration des pertes de retour d'écho)

$NGE(n)$: Rapport des carrés des normes des gains

$NM(n)$: Normalized Misalignment (Norme du vecteur d'erreur des coefficients normalisée)

$Ndg(n)$: Norme de la différence des gains

Liste des tableaux et figures

• Tableaux

I.1: Résumé de l'algorithme NLMS	19
I.2: Résumé de l'algorithme RLS	21
I.3: Résumé de l'algorithme FRLS	27
II.1: Résumé d'algorithmes FRLS Numériquement Stables (NS-FRLS)	39
III.1: Matrices \mathbf{F}_{22} et \mathbf{F}_{33} pour le choix C_1 ($\mu^\gamma = \mu^\beta = \mu^b = 1$); $0 \leq \mu_s \leq 1$	58
III.2: Matrices \mathbf{F}_{22} et \mathbf{F}_{33} pour le choix C_2 ($\mu^\gamma = 0, \mu^\beta = \mu^b = 1$); $0 \leq \mu_s \leq 1$	58
III.3: Matrices \mathbf{F}_{22} et \mathbf{F}_{33} pour le choix C_3 ($\mu^\gamma = 1, \mu^\beta = 1.5, \mu^b = 0.5$); $0 \leq \mu_s \leq 1$	59
III.4: Matrices \mathbf{F}_{22} et \mathbf{F}_{33} pour le choix C_4 ($\mu^\gamma = 0, \mu^\beta = 1.5, \mu^b = 0.5$); $0 \leq \mu_s \leq 1$	59
III.5: Comportement des algorithmes FRLS pour un ordre L par rapport à AR($M=2m$) et k_i ...	80
IV.1: Résumé de l'algorithme M-SMFTF ($6L$)	93
IV.2: Résumé de l'algorithme RM-SMFTF ($2L+4P$; $P \ll L$)	95
V.1 : Complexité de calcul et Condition de stabilité des algorithmes étudiés	121

• Figures

I.1 : Communication bidirectionnelle entre deux salles.	10
I.2 : Principe de l'annulation d'écho acoustique.	11
I.3: Identification linéaire.	14
I.4: Identification par filtrage adaptatif.	15
II.1: Evolution de la variable de Vraisemblance γ_n et de l'Indicateur de Divergence ξ_n	37
II.2: Evolution de l'indicateur des erreurs numériques $\sigma_\xi^2(n)$	38
III.1: Système de filtrage FIR adaptatif.	63
III.3a: Evolution des erreurs numériques $\sigma_\xi^2(n)$ de la version V_1 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)	67
III.3b: Evolution des erreurs numériques $\sigma_\xi^2(n)$ de la version V_2 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)	67
III.3c: Evolution des erreurs numériques $\sigma_\xi^2(n)$ de la version V_3 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)	68
III.3d: Evolution des erreurs numériques $\sigma_\xi^2(n)$ de la version V_4 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)	68
III.4: Comparaison de l'évolution des erreurs numériques $\sigma_\xi^2(n)$ des meilleures versions pour $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)	69

III.5: Comparaison de l'évolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version C_3V_1 avec les versions V_B et V_S pour $L=32$, $\lambda=0.9896$ ($p=3$)	69
III.6: Evolution des variables γ_n , ξ_n et $NGE(n)$: (C_3V_1 , $\mu_s=0.5$) ; $L=32$, $p=1.95$	70
III.7: Evolution des variables γ_n , ξ_n et $NGE(n)$: (C_3V_1 , $\mu_s=0.5$) ; $L=32$, $p=3$	71
III.8: Comparaison des courbes théorique et de simulation de $MSE(n)$ pour (a): $L=32$; (b): $L=256$.	73
III.9: Comparaison des courbes théorique et de simulation de $NM(n)$ pour (a): $L=32$; (b): $L=256$.	74
III.10a: Evolution des variables γ_n et ξ_n ($L=2$, k_0 entier)	77
III.10b: Evolution du module des pôles du prédicteurs aller ($L=2$, k_0 entier)	77
III.11a: Evolution des variables γ_n et ξ_n ($L=2$, k_0 non entier)	77
III.11b: Evolution du module des pôles du prédicteurs aller ($L=2$, k_0 non entier)	77
III.12a: Evolution des variables γ_n et ξ_n ($L=3$, k_0 entier)	78
III.12b: Evolution du module des pôles du prédicteurs aller ($L=3$, k_0 entier)	78
III.13a: Evolution des variables γ_n et ξ_n ($L=3$, k_0 non entier)	79
III.13b: Evolution du module des pôles du prédicteurs aller ($L=3$, k_0 non entier)	79
III.14a: Evolution des variables γ_n et ξ_n ($L=3$, k_0 entier) avec $SNR=60dB$.	82
III.14b: Evolution du module des pôles du prédicteurs aller ($L=3$, k_0 entier) avec $SNR=60dB$	82
III.15 : Gain variable.	84
III.16 : Comparaison de la capacité de poursuite du NS-FRLS et du NS-FRLS accéléré	84
IV.1: Système de filtrage FIR adaptatif.	87
IV.2: Evolution de la norme de la différence des gains $Ndg(n)$ - (a): $L=32$ et (b): $L=256$	104
IV.3: Influence de la valeur η sur l'algorithme M-SMFTF - (a): $L=32$ et (b): $L=256$	106
IV.4: Influence de la valeur c_a sur l'algorithme M-SMFTF - (a): $L=32$ et (b): $L=256$	107
IV.5: Influence de la valeur η sur l'algorithme RM-SMFTF - (a): $P=16$ et (b): $P=32$	108
IV.6: Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS - Entrée USASI et $L=32$, (a) : sans bruit en sortie ; (b) : avec bruit en sortie $SNR=50dB$.	110
IV.7: Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS - Entrée USASI et $L=256$, (a) : sans bruit en sortie ; (b) : avec bruit en sortie $SNR=50dB$	111
IV.8 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS - Entrée USASI et $L=32$ avec une non stationnarité du système	112
IV.9 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS - Entrée USASI et $L=256$ avec une non stationnarité du système	112
IV.10: Effet de l'ordre du prédicteur P sur l'algorithme RM-SMFTF - Entrée USASI et $L=256$ avec une non stationnarité du système	113
V.1 : Système d'annulation d'écho	116
V.2a : Réponse impulsionnelle d'un habitacle de voiture ($F_c=16kHz$)	117
V.2b : Réponse impulsionnelle d'une salle de téléconférence ($F_c=16kHz$)	117
V.3: Spectre du modèle AR(20)	119

V.4 :	Signal de parole ($F_c=16\text{kHz}$)	120
V.5:	Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	124
V.6:	Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, avec bruit additif sur la sortie ($SNR=50\text{dB}$)	124
V.7:	Comparaison entre les algorithmes RM-SMFTF et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	125
V.8:	Comparaison entre les algorithmes RM-SMFTF et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, avec bruit additif sur la sortie ($SNR=50\text{dB}$)	125
V.9:	Evolution du <i>MSE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	126
V.10:	Evolution du <i>MSE</i> des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	126
V.11:	Evolution du <i>ERLE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	127
V.12:	Evolution du <i>ERLE</i> des algorithmes RM-SMFTF et NLMS pour un canal non Stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie	127
V.13:	Evolution du <i>MSE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie	128
V.14:	Evolution du <i>MSE</i> des algorithmes RM-SMFTF et NLMS pour un canal non Stationnaire (Habitable de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie	128
V.15:	Evolution du <i>ERLE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie	129
V.16:	Evolution du <i>ERLE</i> des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie	129
V.17:	Evolution du <i>MSE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie	130
V.18:	Evolution du <i>MSE</i> des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie	130
V.19:	Evolution du <i>ERLE</i> des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non	

stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.	131
V.20: Evolution du <i>ERLE</i> des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.	131
V.21: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.	133
V.22: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, avec bruit additif sur la sortie($SNR=50dB$).	133
V.23: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Salle Téléconférence), Entrée non stationnaire (Parole), Taille du filtre $L=1500$, sans bruit additif sur la sortie.	134
V.24: Comparaison entre les algorithmes RM-SMFTF NLMS pour un canal non stationnaire (Salle Téléconférence), Entrée non stationnaire (Parole), Taille du filtre $L=1500$, sans bruit additif sur la sortie.	134
V.25: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.	135
V.26: Comparaison entre les algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.	135

INTRODUCTION GENERALE

Les moyens de communication comme le téléphone et les réseaux informatiques sont un élément essentiel des télécommunications à longue distance. La conversation téléphonique permet déjà un haut niveau de communication par le choix des mots et le ton de la voix, la visiophonie ajoute de nouvelles dimensions à cette communication : expression du visage, gestes, langage du corps, environnement visible. Des documents avec des textes, des images ou autres données peuvent être utilisés pour venir appuyer le face à face, dans le but d'une meilleure compréhension. La téléconférence est une technique permettant d'établir la communication entre deux salles de réunions. La prise de son dans chacune des salles considère l'orateur comme la source utile. Malheureusement, tous les équipements mains libres classiques ont un problème de réaction acoustique qui dépend de la disposition du haut parleur et du microphone, de leur environnement immédiat, et de l'endroit où ils sont utilisés. Le microphone mains libres n'est pas seulement exposé aux sons diffusés directement par l'interlocuteur proche. Une partie du son issue du haut parleur, revient directement au microphone à travers l'air, en raison de la réflexion des ondes sonores par les objets environnants et puis est renvoyée à l'interlocuteur distant. Ce problème est rencontré dans la plupart des systèmes de communication entre deux locuteurs. Il est d'autant plus marqué quand il s'agit de mains-libres. La simulation est la suivante : le signal émis par un locuteur dit « distant, lointain » est transmis vers un locuteur dit « local, proche », disposant d'un terminal de communication mains-libres. Il est diffusé par le haut-parleur, puis capté par le microphone de ce terminal. Il est alors renvoyé, via le réseau de télécommunications, vers le locuteur distant, qui entend, avec un certain délai, son propre signal, dénommé écho. Ceci crée une gêne : elle doit être réduite par un système d'annulation d'écho, dont le but est de reproduire le couplage haut-parleur/microphone et ainsi d'obtenir une estimation de l'écho, qui sera soustraite de l'observation microphonique.

L'objectif de notre travail est de développer et d'aboutir à des algorithmes efficaces pour des applications d'annulation d'écho acoustique, donc de filtrage adaptatif. Le choix de l'algorithme se fera en fonction des critères suivants:

- la rapidité de convergence qui sera le nombre d'itérations nécessaires pour converger de la solution optimale ;
- la capacité de poursuite des variations (non stationnarités) du système ;
- la robustesse au bruit ;
- la complexité ;
- les propriétés numériques (stabilité et précision) dans le cas d'une précision limitée sur les données et les coefficients du filtre.

Nous menons donc notre étude pour améliorer les performances de ce système et de diminuer sa complexité de calcul au maximum, tout en maintenant un équilibre entre son coût d'implantation et sa qualité. Les algorithmes utilisés dans le système d'annulation d'écho acoustique représentent une charge de calcul trop importante qu'il faut chercher à réduire. Pour cela, une étude théorique a porté sur les algorithmes d'adaptation utilisés dans le système d'annulation d'écho acoustique pour nous fournir une bonne connaissance du système et nous permettre de proposer des améliorations entraînant une réduction de la complexité de calcul et des performances adaptatives satisfaisantes.

Au chapitre I et dans un premier temps, nous posons le problème de l'écho acoustique et les façons d'agir pour le résoudre. La solution retenue est d'utiliser un système d'annulation d'écho acoustique qui devrait idéalement supprimer tout ce qui provient du haut-parleur du système de diffusion. Ce système consiste à utiliser un filtre qui génère une estimée de l'écho réel. La différence entre l'écho réel et son estimée est l'écho résiduel, ou encore l'erreur d'estimation. Le but est de minimiser l'erreur d'estimation pour réduire au mieux l'écho. L'annulation d'écho acoustique peut donc être abordée comme un problème de minimisation. Dans un deuxième temps, notre objectif est de diminuer progressivement l'écart entre le filtre optimal et le filtre adaptatif. Deux familles d'approches peuvent être envisagées : l'une fondée sur la méthode du gradient stochastique (Least Mean Squares : LMS), l'autre sur l'algorithme de moindres carrés récursifs (Recursive Least Squares : RLS).

L'algorithme RLS, basé sur les moindres carrés, possède la caractéristique très intéressante de converger plus rapidement que le gradient stochastique (LMS). Cette convergence rapide se fait au détriment de la complexité de calcul. Le FRLS, version rapide du RLS, basé également sur les moindres carrés, allie la rapidité de convergence du RLS et une complexité de calcul réduite proche du LMS. Cependant, une sérieuse limitation du FRLS est causée par un problème

d'instabilité numérique. Une certaine redondance nous permet d'avoir un contrôle sur l'erreur générée par la troncature numérique, qui fera l'objet du chapitre II. Le principe de base de la méthode de stabilisation, est de calculer les équations de propagation des erreurs dans les algorithmes FRLS puis de modifier, par une contre réaction convenable, les propriétés numériques de l'algorithme ou plus précisément certaines de ces variables récursives dont l'instabilité numérique est connue ou établie. Une étude complète des erreurs d'arrondi de l'algorithme adaptatif passe par l'étude de la propagation des erreurs, les propriétés du bruit d'arrondi et l'accumulation des erreurs. En d'autres termes, comment les erreurs générées à un instant se propagent dans les récursions suivantes, description de l'erreur quand elle est générée à un instant donné et comment les erreurs à différents instants interagissent et s'accumulent.

Au chapitre III, nous présentons l'analyse de la propagation des erreurs numériques de la partie prédiction des algorithmes FRLS modifiés par leurs nouvelles définitions que nous appelons algorithmes des moindres carrés récursifs rapides numériquement stables (NS-FRLS). Après la modification de l'algorithme et le calcul des nouvelles équations de propagation des erreurs numériques, nous étudions comment éviter l'instabilité numérique. Nous montrons, pour un signal stationnaire en entrée, que ces algorithmes ainsi modifiés sont stables numériquement pour un choix convenable de paramètres et une condition sur le facteur d'oubli. Nous présentons ensuite une étude détaillée pour des signaux parfaitement prédictibles, cas des sinusoides pures. Nous proposons, afin d'éviter l'instabilité de l'algorithme NS-FRLS, un choix de la fréquence d'échantillonnage par rapport à la fréquence de la sinusoïde.

Les algorithmes utilisés dans le système d'annulation d'écho acoustique représentent une charge de calcul trop importante (NS-FRLS, $8L$) qu'il faut chercher à réduire. Pour cela, une étude théorique a porté sur les algorithmes d'adaptation utilisés dans le système d'annulation d'écho acoustique pour nous fournir une bonne connaissance du système et nous permettre de proposer quelques améliorations entraînant une réduction satisfaisante de calcul qui fera l'objet du chapitre IV. Nous proposons ainsi un nouvel algorithme rapide (M-SMFTF) pour le filtrage adaptatif. L'algorithme M-SMFTF, simplifié et modifié de type FTF, où le gain d'adaptation est obtenu seulement à partir des variables de la prédiction aller et, en employant une nouvelle méthode récursive pour calculer la variable de vraisemblance. Cet algorithme présente un intérêt certain, pour l'adaptation de filtres très longs comme ceux utilisés dans les problèmes d'annulation d'écho acoustique, à cause de sa complexité réduite, sa stabilité numérique et sa convergence en présence du signal de parole. La complexité de calcul de cet algorithme est de $6L$

et celle-ci est considérablement réduite à $(2L+4P)$ quand on utilise un prédicteur aller réduit d'ordre P ($P \ll L$). Les analyses des propriétés numériques de l'algorithme M-SMFTF montrent la supériorité en termes de capacité de poursuite à l'algorithme NS-FRLS.

Au chapitre V, nous comparons les performances adaptatives de l'algorithme M-SMFTF ($6L$) et l'algorithme à prédicteur réduit RM-SMFTF ($2L+4P$) avec les algorithmes NS-FRLS ($8L$) et NLMS ($2L$). Nous nous plaçons dans un contexte d'annulation d'écho acoustique. La comparaison sera faite dans plusieurs situations. Les simulations sont effectuées lorsque le signal d'excitation est stationnaire, très corrélé et où le système à identifier est non stationnaire. Nous évaluons par la suite, les performances adaptatives de l'algorithme dans des situations réelles où le signal d'entrée est de la parole avec un bruit additif en sortie. L'algorithme M-SMFTF surpasse les algorithmes adaptatifs classiques en raison de sa vitesse de convergence qui approche celle de l'algorithme RLS et de sa complexité de calcul qui est légèrement plus grande que celle de l'algorithme NLMS.

Chapitre I

IDENTIFICATION PAR ALGORITHMES ADAPTATIFS

I.1 Introduction

Le progrès technologique dans le domaine des télécommunications nous permet d'échanger de l'information sans avoir à se déplacer physiquement pour rencontrer son correspondant. Il permet d'avoir plusieurs types de terminaux, parmi lesquels les terminaux audio pour la communication de la parole prennent une place assez importante. Des exemples typiques de ces terminaux audio sont le téléphone mains-libre et les terminaux de groupe pour la téléconférence. Lors de la mise en place d'une communication bidirectionnelle entre deux salles (Fig. I.1), une boucle de transmission fermée est établie. Le signal émis par la salle distante est réémis vers cette même salle à cause du couplage existant entre le haut-parleur et le microphone de prise de son de la salle locale. Alors, si la transmission introduit un retard important (de l'ordre de plusieurs millisecondes), les personnes présentes dans la salle distante réentendent leurs propres voix ; c'est le phénomène d'écho acoustique [1, 2]. Ce dernier dû au canal acoustique de couplage qui, par définition, représente la transformation du signal original diffusé par le haut-parleur et capté par les microphones de prise de son [3].

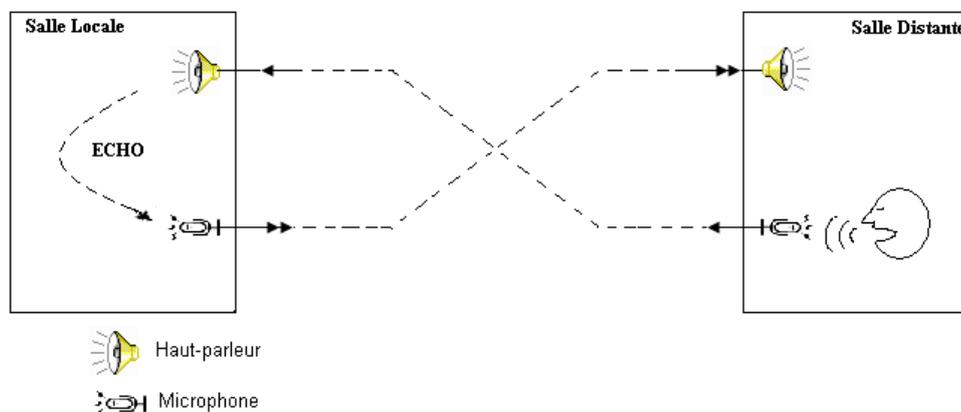


Fig. I.1 : Communication bidirectionnelle entre deux salles

Pour résoudre le problème posé par l'écho acoustique, il est possible d'agir de deux façons :

La première est de réduire le couplage acoustique, c'est-à-dire, combattre les causes de l'écho acoustique par des moyens physiques tels que l'utilisation de microphones directifs, l'éloignement des haut-parleurs et des microphones, le traitement acoustique de la salle par l'utilisation de revêtements absorbants, ou encore l'utilisation de moyens électroniques comme l'incorporation de commutateurs et de gains variables dans la chaînes de transmission [1]. Ces solutions sont toutefois limitées soit à cause de leur rigidité et de leur coût dans le cas des solutions physiques, soit parce qu'elles ne permettent pas une communication bidirectionnelle confortable avec l'utilisation des commutateurs électroniques.

La seconde façon est de traiter le signal capté par le microphone, c'est-à-dire, utiliser un système d'annulation d'écho acoustique (Fig. I.2) [4, 5] qui devrait idéalement supprimer tout ce qui provient du haut-parleur du système de diffusion.

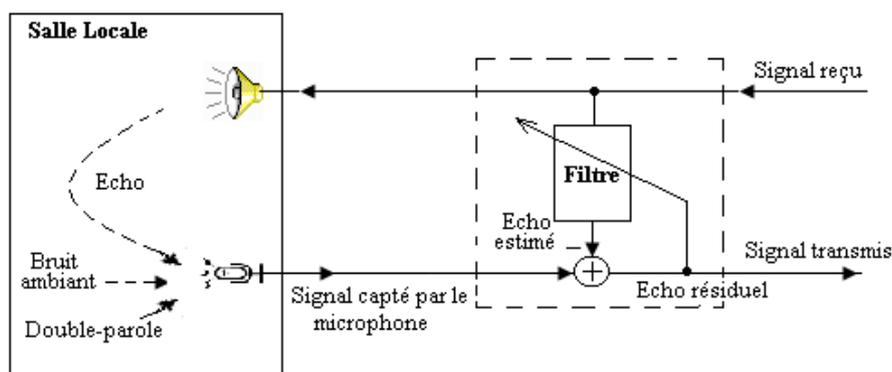


Fig. I.2 : Principe de l'annulation d'écho acoustique

Le microphone capte à la fois la parole du locuteur proche (double-parole) et la parole du locuteur distant (signal écho), à cause du couplage acoustique. Un système d'annulation d'écho estime l'écho acoustique entre l'entrée (signal envoyé dans le haut-parleur) et la sortie (signal récupéré par le microphone), et il retranche cette estimation du signal microphonique, réalisant ainsi idéalement l'annulation de l'écho sans affecter le signal local de parole. En effet, l'annulation de cet écho acoustique est l'estimation du canal acoustique entre le haut-parleur et le microphone par des techniques d'identification de systèmes linéaires, suivies d'un traitement correctif approprié. Le modèle est linéaire, car le canal acoustique est en première approximation composé essentiellement de retards et d'atténuations [6, 7].

L'annulation d'écho acoustique consiste à utiliser un filtre qui génère une estimée de l'écho réel. En effet, le couplage acoustique est un phénomène complexe, difficilement réductible à petit nombre de paramètres. L'approche la plus directe, mais aussi la plus lourde pour le présenter, est la réponse impulsionnelle mesurée entre l'accès électrique au haut-parleur et la sortie du microphone. La réponse impulsionnelle estimée doit être suffisamment longue pour que l'écho estimé soit proche de l'écho vrai. Il convient de noter que d'autres modèles du couplage ont été proposés dans le but de réduire le nombre des paramètres. Certains se révèlent insuffisants et difficiles à identifier (réponses de type « retards plus échos », modèles de réverbérateurs), et d'autre part les modèles à filtres à réponse impulsionnelle infinie (IIR) [8] ne semblent pas adaptés au phénomène physique de l'écho, et ne donnent pas actuellement des améliorations suffisantes pour justifier l'utilisation d'un modèle plus compliqué que le modèle simple à réponse impulsionnelle finie (FIR) [9].

La différence entre l'écho réel et son estimée est l'écho résiduel, ou encore l'erreur d'estimation. Le but est de minimiser l'erreur d'estimation pour réduire au mieux l'écho. L'annulation d'écho acoustique peut donc être abordée comme un problème de minimisation. Un filtre optimum selon la théorie de Wiener minimise l'erreur quadratique moyenne. Ce filtre optimum dépend de l'environnement, notamment du signal d'entrée et de l'écho généré dans la salle locale. Or le signal d'entrée et le signal d'écho ne sont pas stationnaires puisque ce sont des signaux de parole. De plus, l'écho dépend également du canal acoustique qui n'est pas invariant dans le temps (mouvements des personnes présentes dans la salle, du déplacement d'objets). Par conséquent, un filtre optimum au sens de Wiener doit être calculé à chaque modification de l'environnement. Un filtre adaptatif évite d'avoir à recalculer un filtre de Wiener à chaque changement de l'environnement. Un tel filtre se distingue d'un filtre conventionnel par ses coefficients qui sont continuellement modifiés par un algorithme d'adaptation pour tendre vers le minimum de l'erreur quadratique moyenne, et pour poursuivre ce minimum en cas de changements externes.

Les algorithmes d'adaptation se caractérisent par la vitesse de convergence des coefficients vers la solution optimum, et par leur complexité. Ce traitement qui consiste en l'annulation de l'écho acoustique, est une application typique du filtrage adaptatif. Cette application pose un problème difficile, car à contraintes contradictoires, pour lequel on cherche encore une solution satisfaisante, à tel point que le problème de l'annulation d'écho acoustique est un sujet de recherche très actif [10 - 15].

I.2 Identification par algorithmes adaptatifs

Les algorithmes adaptatifs sont un puissant moyen pour parvenir à optimiser un système de structure spécifiée en fonction d'un environnement donné. 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.

Nous rappelons les modèles de filtrage linéaire usuels en signal et les algorithmes les plus classiques pour estimer ces modèles. Très souvent les grandeurs observées sont constituées par deux flots aléatoires. Une suite $\{x_n\}$ d'échantillons d'entrée dépendant du temps n , qui est traitée par le filtre W à optimiser. Une suite temporelle $\{d_n\}$ qui constitue une référence bruitée du signal désiré. L'algorithme d'adaptation du filtre W , est contrôlé par la suite des erreurs

$$\tilde{\varepsilon}_n = d_n - \hat{y}_n \quad (\text{I.1a})$$

où

$$\hat{y}_n = W[x_n] \quad (\text{I.1b})$$

est le signal estimé. Le filtre estimé W , dont l'état est le plus souvent calculé à l'instant $(n-1)$, représenté par w_{n-1} , c'est-à-dire sur la base des observations (x_i, d_i) jusqu'à l'instant $i = n-1$ à l'exclusion de $i \geq n$. Une telle erreur est appelée « a priori » car la nouvelle entrée x_n y est traitée à l'instant n par le filtre w_{n-1} non encore adapté. On peut aussi utiliser l'erreur « a posteriori » recalculée selon (I.1a) mais après avoir incrémenté w_{n-1} en w_n [16]. Que l'erreur soit prise a priori ou a posteriori, l'objectif est toujours d'amener le signal estimé \hat{y}_n à être proche de l'estimation du signal y_n . Nous supposons que le signal désiré du modèle est :

$$d_n = y_n + z_n \quad (\text{I.2a})$$

où

$$y_n = H[x_n] \quad (\text{I.2b})$$

est approché par un traitement linéaire du signal d'entrée x_n , et où H est un filtre linéaire inconnu. Le signal z_n composé d'un bruit v_n , supposé indépendant de x_n , et du signal utile s_n . On a alors affaire à un problème d'identification linéaire où le filtre estimé W doit s'approcher de H . Ceci est illustré par le schéma suivant (Fig. I.3) :

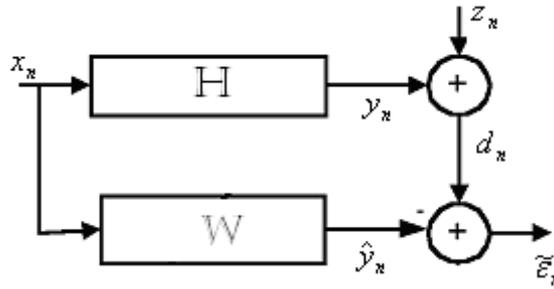


Fig. I.3: Identification linéaire

La qualité de W est contrôlée par son erreur de sortie $\tilde{\varepsilon}_n$. En absence du signal utile ($s_n = 0$), l'erreur de contrôle s'écrit :

$$\tilde{\varepsilon}_n = v_n + \zeta_n \quad (\text{I.3a})$$

avec

$$\zeta_n = H[x_n] - W[x_n] \quad (\text{I.3b})$$

L'erreur de contrôle apparaît comme un estimateur du bruit v_n du modèle (I.2a). De plus comme v_n est indépendant de x_n , les deux composantes ζ_n et v_n de (I.3a) sont indépendantes. Ainsi choisir l'estimation W de H qui minimise (en puissance) l'erreur de sortie $\tilde{\varepsilon}_n$, c'est aussi minimiser l'erreur d'estimation ζ_n sur le signal y_n .

En effet si x_n et v_n sont des suites stationnaires, alors l'erreur (I.3a) l'est aussi si le filtre W ne varie pas. L'erreur sera non stationnaire par la variation du filtre W ($\mathbf{w}_{n-1} \rightarrow \mathbf{w}_n$).

Pour que l'écart entre le filtre H et son estimation soit le plus faible possible, la structure du filtre linéaire W doit être adaptée au problème. Dans un contexte non stationnaire, le modèle linéaire (I.2a) devient évolutif. Les changements au cours du temps peuvent affecter la statistique de x_n , celle de z_n ou encore le filtre linéaire H .

Dans la structure transverse, le signal estimé est donné par la convolution suivante [17-19]:

$$\hat{y}_n = \mathbf{w}_{n-1}^T \mathbf{x}_n \quad (\text{I.4})$$

où

$$\mathbf{x}_n = [x_n, x_{n-1}, \dots, x_{n-L+1}]^T \quad (\text{I.4a})$$

désigne un vecteur qui résume le passé du signal d'entrée x_n sur une durée de L échantillons et

$$\mathbf{w}_n = [w_{1,n}, w_{2,n}, \dots, w_{L,n}]^T \quad (\text{I.4b})$$

est directement une estimation de la réponse impulsionnelle de H , qui est dit à réponse impulsionnelle finie (FIR). Le filtre H pourrait aussi être à réponse impulsionnelle infinie (IIR) mais approché par une estimation tronquée de sa réponse. Elle comporte une série de paramètres à optimiser. Il s'agit de mesurer l'adéquation de l'estimateur W au modèle du filtre H . L'exposant T désigne l'opérateur de transposition. Deux principaux critères sont utilisés, les moindres carrés exacts et les moindres carrés moyens.

Pour des signaux aléatoires, x_n et v_n , stationnaires et ergodiques le critère des moindres carrés exacts s'identifie avec le critère des moindres carrés moyens lorsque n devient très grand, ceci en conséquence de la loi des grands nombres qui assimile moyennes d'ensemble et moyennes temporelles.

Les algorithmes adaptatifs permettent de minimiser récursivement les deux critères précédents en recherchant le filtre w_n , selon

$$w_n = w_{n-1} + f[\bar{\varepsilon}_n, x_n] \quad (I.5)$$

Ce sont des algorithmes basés principalement sur l'erreur d'estimation du filtre à l'instant n . Les coefficients de la réponse impulsionnelle du filtre sont adaptés en fonction de l'erreur par une boucle de retour comme le montre la Fig. I.4. Cette adaptation nécessite une séquence d'apprentissage et une stratégie de mise à jour des coefficients du filtre dont l'objectif est la minimisation d'une erreur.

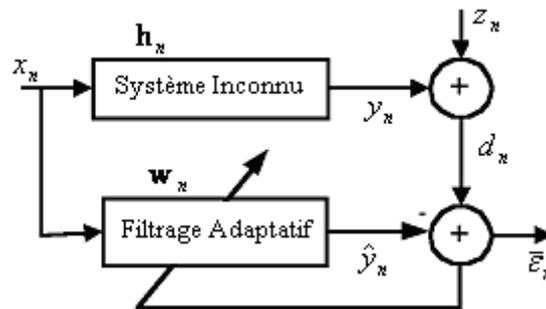


Fig. I.4: Identification par filtrage adaptatif

L'erreur de filtrage a priori $\bar{\varepsilon}_n$ à l'instant n , calculée avant la mise à jour du filtre, est donnée par :

$$\bar{\varepsilon}_n = d_n - w_{n-1}^T x_n \quad (I.6)$$

Le principal rôle d'un algorithme adaptatif est d'ajuster un vecteur de paramètres w_n pour un objectif bien défini. La qualité de w_n est contrôlée et adaptée par son erreur de sortie $\bar{\varepsilon}_n$. Les

algorithmes de poursuite donnent au filtre \mathbf{w}_n la capacité de s'approcher de la vraie valeur \mathbf{h}_n du filtre H, à partir d'une valeur initiale \mathbf{w}_0 . Puis, en régime permanent \mathbf{w}_n se met à fluctuer autour de la valeur vraie \mathbf{h}_n , qui est évolutive.

I.3 Présentation des algorithmes

L'objectif de l'algorithme adaptatif est de fournir à chaque instant n une estimée la plus fidèle possible du système à identifier. Cette estimation est réalisée en minimisant une fonction coût, notée $J_n(\mathbf{w})$, qui dépend des coefficients du filtre adaptatif et plus généralement de l'erreur en sortie du filtre. Nous allons voir dans les prochains paragraphes les algorithmes qui découlent de ces différentes fonctions de coût.

La fonction coût à minimiser est l'erreur quadratique moyenne en sortie du filtre qui s'écrit [20]:

$$J_n(\mathbf{w}) = E\left\{\left(d_n - \mathbf{w}_n^T \mathbf{x}_n\right)^2\right\} \quad (\text{I.7})$$

où $E\{\cdot\}$ désigne l'espérance mathématique. L'erreur quadratique moyenne est une fonction du second degré du vecteur \mathbf{w}_n . La représentation graphique de $J_n(\mathbf{w})$ en fonction des coefficients du filtre est une surface parabolôïde qui a un minimum unique. Les coordonnées du minimum correspondent aux coefficients du filtre optimum au sens de Wiener. Si on suppose les signaux x_n et d_n centrés stationnaires, la nullité du gradient de la fonctionnelle $J_n(\mathbf{w})$ est donnée par [21, 22]:

$$\nabla_{\mathbf{w}} J_n(\mathbf{w}) = 0 \Leftrightarrow 2\mathbf{R}_{xx} \mathbf{w}_n - 2\mathbf{p}_{xd} = 0 \quad (\text{I.8})$$

où

$$\mathbf{R}_{xx} = E\left\{\mathbf{x}_n \mathbf{x}_n^T\right\} \quad (\text{I.9})$$

est la matrice d'autocorrélation du vecteur d'entrée \mathbf{x}_n , et où

$$\mathbf{p}_{xd} = E\left\{\mathbf{x}_n d_n\right\} \quad (\text{I.10})$$

est le vecteur d'intercorrélacion entre le signal désiré d_n et le signal d'entrée du filtre adaptatif.

Si la matrice d'autocorrélation est inversible, l'expression \mathbf{w}_{opt} du filtre optimal au sens de l'erreur quadratique moyenne ou filtre de Wiener est donnée par [23-26] :

$$\mathbf{w}_{\text{opt}} = \mathbf{R}_{xx}^{-1} \mathbf{p}_{xd} \quad (\text{I.11})$$

Si nous posons \mathbf{h} le vecteur des L premiers coefficients du système à identifier, et y_n la réponse à ce système où

$$y_n = \mathbf{h}^T \mathbf{x}_n \quad (\text{I.12})$$

On montre aisément que l'on obtient l'égalité suivante :

$$\mathbf{w}_{\text{opt}} = \mathbf{h} \quad (\text{I.13})$$

L'utilisation de l'équation (I.11) exige la connaissance des propriétés statistiques de \mathbf{R}_{xx} et de \mathbf{p}_{xd} . Lorsque ces deux quantités ne sont pas disponibles, nous utilisons des observations successives pour les estimer. Nous effectuons ces estimations à partir des données ce qui ne conduit pas au système optimal, mais donne de bons résultats en pratique.

Dans la suite, notre objectif est de diminuer progressivement l'écart entre le filtre optimal \mathbf{w}_{opt} et le filtre adaptatif \mathbf{w}_n , par itérations successives, en partant de valeurs initiales quelconques pour les paramètres du filtre adaptatif.

Deux familles d'approches peuvent être envisagées : l'une fondée sur la méthode du gradient stochastique (Least Mean Squares : LMS), l'autre sur l'algorithme de moindres carrés récursifs (Recursive Least Squares : RLS).

I.3.1 Famille du gradient stochastique (LMS)

La famille d'algorithmes du gradient stochastique (LMS) est particulièrement populaire dans les applications compte tenu de sa simplicité de mise en œuvre, bien qu'elle découle d'une approximation, à savoir approcher une moyenne d'ensemble d'une variable aléatoire par sa valeur instantanée.

Les coefficients du filtre sont ajustés à chaque instant n par un algorithme d'adaptation itératif. Une procédure itérative bien connue qui annule le gradient de l'erreur quadratique moyenne (*MSE*) est donnée dans [23]. L'algorithme du gradient déterministe s'écrit :

$$\mathbf{w}_n = \mathbf{w}_{n-1} - \mu [\nabla_w E\{\bar{\varepsilon}_n^2\}] \quad (\text{I.14})$$

où μ est le pas d'adaptation qui contrôle la convergence de l'algorithme.

En développant la fonction $\nabla_w E\{\bar{\varepsilon}_n^2\}$, la relation (I.14) devient :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu E\{\mathbf{x}_n \bar{\varepsilon}_n\} \quad (\text{I.15})$$

L'idée des algorithmes de type gradient stochastique est de remplacer la moyenne statistique dans l'algorithme du gradient déterministe d'équation (I.15) par sa valeur instantanée. La mise à jour des paramètres du filtre adaptatif devient ainsi :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu \mathbf{x}_n \bar{\varepsilon}_n \quad (\text{I.16})$$

La convergence de l'algorithme LMS est garantie sous la condition suivante [23, 24]:

$$0 < \mu < \frac{1}{\lambda_{\max}} \quad (\text{I.17a})$$

où λ_{\max} est la plus grande valeur propre de la matrice d'autocorrélation \mathbf{R}_{xx} donnée par (I.5).

En pratique, on utilise souvent, pour le choix de la constante μ , une condition plus restrictive [17, 18, 27-30]:

$$0 < \mu \leq \frac{1}{\text{tr}(\mathbf{R}_{xx})} = \frac{1}{L\sigma_x^2} \quad (\text{I.17b})$$

où $\text{tr}(\cdot)$ désigne trace d'une matrice et σ_x^2 représente la variance du signal d'entrée. Cette dernière condition exploite le caractère défini positif de la matrice d'autocorrélation.

Les performances asymptotiques de l'algorithme LMS peuvent se résumer par le compromis à faire d'une part à l'aide d'un *MSE* faible (en précision finie) c'est-à-dire un pas faible, et d'autre part une bonne capacité de poursuite pour les signaux non stationnaires à l'entrée, ce qui demande une grande valeur du pas. Nous nous assurons que la valeur du pas μ ne soit ni trop grande, ce qui pourrait entraîner une divergence de l'algorithme, ni trop faible pour suivre correctement les variations du système à identifier. Lorsque le signal est non stationnaire, sa puissance varie dans le temps. Il est donc difficile de fixer un μ optimal pour toute la durée d'adaptation. En effet, la condition de convergence de l'algorithme LMS dépend de la puissance du signal d'entrée. Pour rendre les performances de l'algorithme indépendantes de cette puissance, nous normalisons le gain scalaire μ de l'algorithme LMS par une quantité dépendante de la puissance du signal d'entrée.

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_n [19], la puissance du signal est :

$$\pi_{x,n} = \frac{\mathbf{x}_n^T \mathbf{x}_n}{L} \quad (\text{I.18a})$$

Pour réduire la complexité calculatoire, on peut éviter à chaque itération la puissance du signal d'entrée $\pi_{X,n}$. Cet algorithme repose sur une méthode récursive pour estimer la puissance du signal d'entrée par la relation suivante [32] :

$$\pi_{X,n} = (1-\gamma)\pi_{X,n-1} + \gamma x_n^2 \quad (\text{I.18b})$$

où le paramètre $\gamma \approx 1/L$.

Cependant, lorsque le vecteur d'entrée prend des valeurs faibles, des problèmes de stabilité peuvent apparaître dus à la précision finie des calculs numériques. Pour cette raison, on introduit un scalaire de régulation c_0 . La condition de stabilité de cet algorithme est

$$0 < \mu < 2 \quad (\text{I.19})$$

et la convergence la plus rapide est obtenue pour $\mu = 1$ [31].

L'algorithme NLMS, résumé à la Table I.1, est de complexité $2L$ (mesurée en nombre de multiplications par itération).

Initialisation: $\mathbf{w}_0 = \mathbf{0}_L$;
Condition de convergence : $0 < \mu < 2$;
Variables disponibles à l'instant n : \mathbf{w}_{n-1} ;
Nouvelles informations : x_n, d_n ;
Erreur de filtrage: $\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n$;
Puissance du signal d'entrée : $\pi_{X,n} = (1-\gamma)\pi_{X,n-1} + \gamma x_n^2$;
Adaptation du filtre : $\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\mu}{L\pi_{X,n} + c_0} \mathbf{x}_n \bar{\varepsilon}_n$

Table I.1: Résumé de l'algorithme NLMS ($2L$)

I.3.2 Famille d'Algorithmes des moindres carrés récursifs (RLS)

I.3.2.1 Algorithme des moindres carrés récursifs (RLS)

L'algorithme des moindres carrés est généralement présenté comme étant la méthode qui, à partir de L points, est capable de déterminer une fonction dont la courbe passe par ces L points. Le critère de minimisation ne fait plus intervenir l'espérance de l'erreur quadratique. La méthode consiste à considérer la fonction coût dites des moindres carrés qui vise à minimiser par rapport

au vecteur des paramètres \mathbf{w}_n , à chaque instant n , un critère défini sur les erreurs pondérées commises depuis l'instant initial. Ce critère est donné par [30] :

$$J_n(\mathbf{w}) = \sum_{i=1}^n \lambda^{n-i} (d_i - \mathbf{w}_n^T \mathbf{x}_i)^2 \quad (\text{I.20})$$

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_n et d_n sont nuls avant l'instant initial $n = 0$. La solution qui exprime la nullité du gradient de la fonctionnelle $J_n(\mathbf{w})$:

$$\nabla_{\mathbf{w}} J_n(\mathbf{w}) = 0 \Leftrightarrow -2 \left[\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i d_i - \left(\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T \right) \mathbf{w}_n \right] = 0 \quad (\text{I.21})$$

est donnée par :

$$\mathbf{R}_n \mathbf{w}_n = \mathbf{p}_n \quad (\text{I.22})$$

et par conséquent, on a

$$\mathbf{w}_n = \mathbf{R}_n^{-1} \mathbf{p}_n \quad (\text{I.23})$$

où la matrice \mathbf{R}_n représente la matrice d'autocorrélation à court terme donnée par :

$$\mathbf{R}_n = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T \quad (\text{I.24})$$

et où le vecteur \mathbf{p}_n désigne un vecteur d'intercorrélation à court terme donné par la relation :

$$\mathbf{p}_n = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i d_i \quad (\text{I.25})$$

L'algorithme (I.23) nécessite l'inversion d'une matrice carrée d'ordre L dont le coût de calcul est de l'ordre L^3 opérations arithmétiques par itération.

En développant les relations (I.24) et (I.25), on déduit leurs relations récursives respectives:

$$\mathbf{R}_n = \lambda \mathbf{R}_{n-1} + \mathbf{x}_n \mathbf{x}_n^T \quad (\text{I.26})$$

$$\mathbf{p}_n = \lambda \mathbf{p}_{n-1} + \mathbf{x}_n d_n \quad (\text{I.27})$$

En manipulant ces dernières, on obtient la solution équivalente à (I.23) mais sous une forme récursive :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{g}_n \bar{\varepsilon}_n \quad (\text{I.28})$$

$$\mathbf{g}_n = \mathbf{R}_n^{-1} \mathbf{x}_n \quad (\text{I.29})$$

où le vecteur \mathbf{g}_n est appelé gain de Kalman, que l'on note

$$\mathbf{k}_n = [k_{1,n}, k_{2,n}, \dots, k_{L,n}]^T \quad (\text{I.30})$$

En appliquant le lemme d'inversion des matrices [30]. L'algorithme des moindres carrés récursifs (Recursive Least Squares : RLS) résout ce problème, avec un nombre d'opérations arithmétiques proportionnel à L^2 . L'inverse de la matrice \mathbf{R}_n s'exprime récursivement grâce à l'équation suivante :

$$\mathbf{R}_n^{-1} = \lambda^{-1} \left[\mathbf{R}_{n-1}^{-1} - \frac{\mathbf{R}_{n-1}^{-1} \mathbf{x}_n \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1}}{\lambda + \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1} \mathbf{x}_n} \right] \quad (\text{I.31})$$

L'algorithme des moindres carrés récursifs (RLS) est résumé à la Table I.2. Notons que, l'inversion de la matrice a été évitée en propageant le gain de Kalman \mathbf{k}_n récursivement au cours du temps. L'algorithme RLS est beaucoup plus performant que l'algorithme NLMS en termes de vitesse de convergence et de précision. Malgré cela, l'algorithme RLS n'est pas très répandu dans les techniques de filtrage adaptatif. La raison principale est son coût calculatoire. C'est pour cette raison que des versions rapides ont été mises au point.

Initialisation: $\mathbf{w}_0 = \mathbf{k}_0 = \mathbf{0}_L$; $\mathbf{R}_0^{-1} = \frac{1}{\mu^2} \mathbf{I}$; $\mu < 1$; \mathbf{I} matrice identité d'ordre L .

Variables disponibles à l'instant n : \mathbf{R}_{n-1}^{-1} , \mathbf{w}_{n-1} ;

Nouvelles informations : x_n , d_n ;

Gain de Kalman : $\mathbf{k}_n = \frac{\mathbf{R}_{n-1}^{-1} \mathbf{x}_n}{\lambda + \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1} \mathbf{x}_n}$;

Erreur de filtrage: $\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n$;

Adaptation du filtre : $\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{k}_n \bar{\varepsilon}_n$;

Propagation de l'inverse de \mathbf{R}_n : $\mathbf{R}_n^{-1} = \lambda^{-1} [\mathbf{R}_{n-1}^{-1} - \mathbf{k}_n \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1}]$

Table I.2: Résumé de l'algorithme RLS (L^2)

I.3.2.2 Algorithmes des moindres carrés récursifs rapides (FRLS)

Plusieurs versions rapides du RLS existent et ont une complexité proportionnelle à L . Les moindres carrés transversaux rapides où le filtre adapté est un filtre transverse. Nous citerons principalement l'algorithme de Kalman rapide (FK) [33] qui est historiquement le premier algorithme rapide, l'algorithme FAEST (Fast a Posteriori Error Sequential Technique) [34] développé indépendamment avec l'algorithme FTF (Fast Transversal Filter) [35].

Dans ce paragraphe nous décrivons brièvement l'essentiel des équations qui permettent d'obtenir les algorithmes des moindres carrés récursifs rapides (Fast RLS : FRLS) à oubli exponentiel, faisant intervenir des erreurs a posteriori [36, 37].

Le but de cette méthode des moindres carrés est de trouver le vecteur \mathbf{w}_n qui minimise le critère (I.20). Une solution récursive à ce problème est donnée par l'algorithme (I.28). 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.29). Si on substitue (I.26) et (I.27) dans (I.22), on obtient une autre forme récursive du vecteur \mathbf{w}_n , strictement équivalente à (I.28), faisant intervenir une erreur de filtrage a posteriori ε_n (calculée après la mise à jour du filtre) et un gain d'adaptation appelé gain de Kalman dual $\tilde{\mathbf{k}}_n$. On a :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \tilde{\mathbf{k}}_n \varepsilon_n \quad (\text{I.32})$$

$$\varepsilon_n = d_n - \mathbf{w}_n^T \mathbf{x}_n \quad (\text{I.33})$$

$$\tilde{\mathbf{k}}_n = \lambda^{-1} \mathbf{R}_{n-1}^{-1} \mathbf{x}_n \quad (\text{I.34})$$

On ne peut pas appliquer l'algorithme (I.32), 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 FRLS est que les erreurs a posteriori se calculent 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 $(L + 1)$:

$$\mathbf{x}_{L+1,n} = \begin{bmatrix} x_n \\ \mathbf{x}_{n-1} \end{bmatrix} \quad (\text{I.35a})$$

$$\mathbf{x}_{L+1,n} = \begin{bmatrix} \mathbf{x}_n \\ x_{n-L} \end{bmatrix} \quad (\text{I.35b})$$

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

La première matrice correspond au vecteur (I.35a) s'écrit :

$$\mathbf{R}_{L+1,n} = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L+1,i} \mathbf{x}_{L+1,i}^T = \sum_{i=1}^n \lambda^{n-i} \begin{bmatrix} x_i \\ \mathbf{x}_{i-1} \end{bmatrix} \begin{bmatrix} x_i & \mathbf{x}_{i-1}^T \end{bmatrix} = \begin{bmatrix} P_n^a & \mathbf{p}_n^{aT} \\ \mathbf{p}_n^a & \mathbf{R}_{n-1} \end{bmatrix} \quad (\text{I.36a})$$

$$P_n^a = \sum_{i=1}^n \lambda^{n-i} x_i^2 \quad (\text{I.36ai}) ; \quad \mathbf{p}_n^a = \sum_{i=1}^n \lambda^{n-i} x_i \mathbf{x}_{i-1} = \lambda \mathbf{p}_{n-1}^a + x_n \mathbf{x}_{n-1} \quad (\text{I.36aii})$$

où l'exposant a , désigne variable aller.

La deuxième matrice correspond au vecteur (I.35b) s'écrit :

$$\mathbf{R}_{L+1,n} = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L+1,i} \mathbf{x}_{L+1,i}^T = \sum_{i=1}^n \lambda^{n-i} \begin{bmatrix} \mathbf{x}_i \\ x_{i-L} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i^T & x_{i-L} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_n & \mathbf{p}_n^b \\ \mathbf{p}_n^{bT} & p_n^b \end{bmatrix} \quad (\text{I.36b})$$

$$p_n^b = \sum_{i=1}^n \lambda^{n-i} x_{i-L}^2 \quad (\text{I.36bi}); \quad \mathbf{p}_n^b = \sum_{i=1}^n \lambda^{n-i} x_{i-L} \mathbf{x}_i = \lambda \mathbf{p}_{n-1}^b + x_{n-L} \mathbf{x}_n \quad (\text{I.36bii})$$

où l'exposant b , désigne variable retour.

Le but des algorithmes FRLS est de propager un vecteur gain. Le calcul du gain de Kalman (I.29) ou du gain de Kalman dual (I.34) 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 [38]. Soit \mathbf{M} une matrice partitionnée :

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{M}_2 \\ \mathbf{M}_3 & \mathbf{M}_4 \end{bmatrix} \quad (\text{I.37})$$

L'inverse de la matrice \mathbf{M} s'écrit de manière générale sous deux formes égales (on suppose la compatibilité des dimensions et l'existence des inverses de certaines matrices de \mathbf{M}):

$$\mathbf{M}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{M}_4^{-1} \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ -\mathbf{M}_4^{-1} \mathbf{M}_3 \end{bmatrix} (\mathbf{M}_1 - \mathbf{M}_2 \mathbf{M}_4^{-1} \mathbf{M}_3)^{-1} \begin{bmatrix} \mathbf{I} & -\mathbf{M}_2 \mathbf{M}_4^{-1} \end{bmatrix} \quad (\text{I.38a})$$

$$\mathbf{M}^{-1} = \begin{bmatrix} \mathbf{M}_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{M}_1^{-1} \mathbf{M}_2 \\ \mathbf{I} \end{bmatrix} (\mathbf{M}_4 - \mathbf{M}_3 \mathbf{M}_1^{-1} \mathbf{M}_2)^{-1} \begin{bmatrix} -\mathbf{M}_3 \mathbf{M}_1^{-1} & \mathbf{I} \end{bmatrix} \quad (\text{I.38b})$$

En utilisant la forme (I.38a) pour inverser la matrice (I.36a) et la forme (I.38b) pour inverser la matrice (I.36b), on aura les expressions suivantes :

$$\mathbf{R}_{L+1,n}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{R}_{n-1}^{-1} \end{bmatrix} + \begin{bmatrix} 1 \\ -\mathbf{R}_{n-1}^{-1} \mathbf{p}_n^a \end{bmatrix} (p_n^a - \mathbf{p}_n^{aT} \mathbf{R}_{n-1}^{-1} \mathbf{p}_n^a)^{-1} \begin{bmatrix} 1 & -\mathbf{p}_n^{aT} \mathbf{R}_{n-1}^{-1} \end{bmatrix} \quad (\text{I.39a})$$

$$\mathbf{R}_{L+1,n}^{-1} = \begin{bmatrix} \mathbf{R}_n^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{R}_n^{-1} \mathbf{p}_n^b \\ 1 \end{bmatrix} (p_n^b - \mathbf{p}_n^{bT} \mathbf{R}_n^{-1} \mathbf{p}_n^b)^{-1} \begin{bmatrix} -\mathbf{p}_n^{bT} \mathbf{R}_n^{-1} & 1 \end{bmatrix} \quad (\text{I.39b})$$

Ce problème est un cas particulier du problème du RLS vu précédemment. Pour sa résolution, il suffit de voir que la forme (I.39a) fait apparaître un prédictor aller optimal au sens des moindres carrés (le terme aller désigne la modélisation de l'échantillon x_n par une combinaison linéaire de son passé $\{x_{n-1}, \dots, x_{n-L}\}$) :

$$\mathbf{a}_n = \mathbf{R}_{n-1}^{-1} \mathbf{p}_n^a \quad (\text{I.40})$$

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

$$J_n(\mathbf{a}) = \sum_{i=1}^n \lambda^{n-i} (x_i - \mathbf{a}_n^T \mathbf{x}_{i-1})^2 \quad (\text{I.41})$$

L'échantillon du signal désiré à l'instant n est x_n et le vecteur de régression est \mathbf{x}_{n-1} . Les versions récursives du prédicteur aller (I.40) faisant intervenir soit le gain de Kalman (I.29), soit le gain de Kalman dual (I.34), s'obtiennent de la même façon que les versions récursives (I.28) et (I.32) du filtre transverse \mathbf{w}_n :

$$\mathbf{a}_n = \mathbf{a}_{n-1} + \mathbf{k}_{n-1} \bar{e}_n \quad (\text{I.42a})$$

$$\mathbf{a}_n = \mathbf{a}_{n-1} + \tilde{\mathbf{k}}_{n-1} e_n \quad (\text{I.42b})$$

où \bar{e}_n et e_n désignent respectivement l'erreur de prédiction aller a priori et l'erreur de prédiction aller a posteriori, qui sont données par :

$$\bar{e}_n = x_n - \mathbf{a}_{n-1}^T \mathbf{x}_{n-1} \quad (\text{I.43})$$

$$e_n = x_n - \mathbf{a}_n^T \mathbf{x}_{n-1} \quad (\text{I.44})$$

De la même façon, la forme (I.39b) fait apparaître un prédicteur retour optimal au sens des moindres carrés (le terme retour désigne la modélisation de l'échantillon x_{n-L} par une combinaison linéaire des échantillons futurs $\{x_n, \dots, x_{n-L+1}\}$) :

$$\mathbf{b}_n = \mathbf{R}_n^{-1} \mathbf{p}_n^b \quad (\text{I.45})$$

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

$$J_n(\mathbf{b}) = \sum_{i=1}^n \lambda^{n-i} (x_{i-L} - \mathbf{b}_n^T \mathbf{x}_i)^2 \quad (\text{I.46})$$

Ce problème est aussi un cas particulier du problème du RLS où échantillon du signal désiré à l'instant n est x_{n-L} et le vecteur de régression est \mathbf{x}_n . Les versions récursives s'obtiennent:

$$\mathbf{b}_n = \mathbf{b}_{n-1} + \mathbf{k}_n \bar{r}_n \quad (\text{I.47a})$$

$$\mathbf{b}_n = \mathbf{b}_{n-1} + \tilde{\mathbf{k}}_n r_n \quad (\text{I.47b})$$

où \bar{r}_n et r_n désignent respectivement l'erreur de prédiction retour a priori et l'erreur de prédiction retour a posteriori, qui sont données par :

$$\bar{r}_n = x_{n-L} - \mathbf{b}_{n-1}^T \mathbf{x}_n \quad (\text{I.48})$$

$$r_n = x_{n-L} - \mathbf{b}_n^T \mathbf{x}_n \quad (\text{I.49})$$

Les expressions $(p_n^a - \mathbf{p}_n^{aT} \mathbf{R}_{n-1}^{-1} \mathbf{p}_n^a)$ et $(p_n^b - \mathbf{p}_n^{bT} \mathbf{R}_n^{-1} \mathbf{p}_n^b)$ dans les expressions (I.39a) et (I.39b) représentent respectivement les variances des erreurs de prédiction aller et retour (minima des critères (I.41) et (I.46)). Elles sont notées α_n et β_n . Leurs versions récursives s'écrivent [37] :

$$\alpha_n = \lambda \alpha_{n-1} + \bar{e}_n e_n \quad (\text{I.50})$$

$$\beta_n = \lambda \beta_{n-1} + \bar{r}_n r_n \quad (\text{I.51})$$

On considère $\mathbf{k}_{L+1,n}^+ = \mathbf{R}_{L+1,n}^{-1} \mathbf{x}_{L+1,n}$, les gains de Kalman étendu à l'ordre $(L+1)$ obtenus en utilisant les expressions (I.39a) et (I.39b) :

$$\mathbf{k}_{L+1,n}^+ = \begin{bmatrix} 0 \\ \mathbf{k}_{n-1} \end{bmatrix} + \frac{e_n}{\alpha_n} \begin{bmatrix} 1 \\ -\mathbf{a}_n \end{bmatrix} \quad (\text{I.52a})$$

$$\mathbf{k}_{L+1,n}^+ = \begin{bmatrix} \mathbf{k}_n \\ 0 \end{bmatrix} + \frac{r_n}{\beta_n} \begin{bmatrix} -\mathbf{b}_n \\ 1 \end{bmatrix} \quad (\text{I.52b})$$

On obtient aussi les deux expressions, $\tilde{\mathbf{k}}_{L+1,n}^+ = \lambda^{-1} \mathbf{R}_{L+1,n-1}^{-1} \mathbf{x}_{L+1,n}$, du gain de Kalman dual étendu à l'ordre $(L+1)$:

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{n-1} \end{bmatrix} + \frac{\bar{e}_n}{\lambda \alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} \quad (\text{I.53a})$$

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} \tilde{\mathbf{k}}_n \\ 0 \end{bmatrix} + \frac{\bar{r}_n}{\lambda \beta_{n-1}} \begin{bmatrix} -\mathbf{b}_{n-1} \\ 1 \end{bmatrix} \quad (\text{I.53b})$$

Si on dispose de la 1^{ière} ou de la $(L+1)$ ^{ième} composante du gain de Kalman étendu ou du gain de Kalman dual étendu, on peut déduire :

$$e_n = \alpha_n k_{1,n}^+ \quad (\text{I.54})$$

$$\bar{e}_n = \lambda \alpha_{n-1} \tilde{k}_{1,n}^+ \quad (\text{I.55})$$

$$r_n = \beta_n k_{L+1,n}^+ \quad (\text{I.56})$$

$$\bar{r}_n = \lambda \beta_{n-1} \tilde{k}_{L+1,n}^+ \quad (\text{I.57})$$

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 l'erreur de filtrage a posteriori (I.33) et remplacer le filtre \mathbf{w}_n par son équation d'adaptation (I.28) il vient :

$$\varepsilon_n = \bar{\varepsilon}_n (1 - \mathbf{k}_n^T \mathbf{x}_n) \quad (\text{I.58})$$

On définit alors la variable de vraisemblance par :

$$\gamma_n = 1 - \mathbf{k}_n^T \mathbf{x}_n = 1 - \mathbf{x}_n^T \mathbf{R}_n^{-1} \mathbf{x}_n \quad (\text{I.59})$$

En théorie, la matrice \mathbf{R}_n et son inverse sont définies positives. Pour un fonctionnement normal de l'algorithme, la variable γ_n doit donc être comprise entre 0 et 1.

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

$$e_n = \gamma_{n-1} \bar{e}_n \quad (\text{I.60})$$

$$r_n = \gamma_n \bar{r}_n \quad (\text{I.61})$$

D'autre part, les équations de mise à jour des vecteurs \mathbf{w}_n , \mathbf{a}_n et \mathbf{b}_n 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 :

$$\mathbf{k}_n = \gamma_n \tilde{\mathbf{k}}_n \quad (\text{I.62})$$

En substituant cette relation dans (I.59) on obtient la relation suivante :

$$\gamma_n = \frac{1}{1 + \tilde{\mathbf{k}}_n^T \mathbf{x}_n} \quad (\text{I.63})$$

Enfin, pour réduire la complexité, il faut calculer récursivement la variable de vraisemblance. Pour cela, calculons la variable de vraisemblance étendu à l'ordre $(L+1)$, $\gamma_n^+ = 1 - \mathbf{x}_{L+1,n}^T \mathbf{k}_{L+1,n}^+$, on obtient deux expressions égales, données respectivement en utilisant (I.52a) et (I.52b) :

$$\gamma_n^+ = \frac{\lambda \alpha_{n-1}}{\alpha_n} \gamma_{n-1} \quad (\text{I.64a})$$

$$\gamma_n^+ = \frac{\lambda \beta_{n-1}}{\beta_n} \gamma_n \quad (\text{I.64b})$$

En posant

$$\theta_n = \frac{\lambda \beta_{n-1}}{\beta_n} \quad (\text{I.65a})$$

et en manipulant les expressions (I.51), (I.56) et (I.57) on aura d'autres formes d'écritures

$$\theta_n = 1 - \bar{r}_n k_{L+1,n}^+ = 1 - \bar{r}_n \gamma_n^+ \tilde{k}_{L+1,n}^+ \quad (\text{I.65b})$$

Ainsi, on calcule alors la variable de vraisemblance d'ordre L par la relation suivante :

$$\gamma_n = \frac{\gamma_n^+}{\theta_n} \quad (\text{I.66})$$

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 proportionnelles à l'ordre L . Les algorithmes FRLS 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. L'algorithme FTF- $7L$, ($7L$ pour la complexité) est résumé dans la Table I.3. Cet algorithme propage le gain de Kalman dual et la variable de vraisemblance.

Initialisation: $\mathbf{w}_0 = \mathbf{a}_0 = \mathbf{b}_0 = \tilde{\mathbf{k}}_0 = \mathbf{0}_L$; $\gamma_0 = 1$; $\alpha_0 = \lambda^L E_0$; $\beta_0 = E_0$; $E_0 \geq \sigma_x^2 \frac{L}{100}$

Variables disponibles à l'instant n : \mathbf{a}_{n-1} ; \mathbf{b}_{n-1} ; $\tilde{\mathbf{k}}_{n-1}$; γ_{n-1} ; α_{n-1} ; β_{n-1} ; \mathbf{w}_{n-1}

Nouvelles informations: x_n , d_n ;

- Partie Prédiction:

Modélisation de: x_n , x_{n-L} ;

$$\bar{e}_n = x_n - \mathbf{a}_{n-1}^T \mathbf{x}_{n-1} ;$$

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} \tilde{\mathbf{k}}_n^+ \\ \tilde{k}_{L+1,n}^+ \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{n-1} \end{bmatrix} + \frac{\bar{e}_n}{\lambda \alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} ;$$

$$\mathbf{a}_n = \mathbf{a}_{n-1} + \bar{e}_n \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} ;$$

$$\alpha_n = \lambda \alpha_{n-1} + \gamma_{n-1} \bar{e}_n^2$$

$$\bar{r}_n = \lambda \beta_{n-1} \tilde{k}_{L+1,n}^+ ;$$

$$\tilde{\mathbf{k}}_n = \tilde{\mathbf{k}}_n^+ + \tilde{k}_{L+1,n}^+ \mathbf{b}_{n-1} ;$$

$$\gamma_n^+ = \frac{\lambda \alpha_{n-1}}{\alpha_n} \gamma_{n-1} ;$$

$$\gamma_n = \frac{\gamma_n^+}{1 - \gamma_n^+ \bar{r}_n \tilde{k}_{L+1,n}^+} ;$$

$$\mathbf{b}_n = \mathbf{b}_{n-1} + \bar{r}_n \gamma_n \tilde{\mathbf{k}}_n ;$$

$$\beta_n = \lambda \beta_{n-1} + \gamma_n \bar{r}_n^2 ;$$

- Partie Filtrage:

$$\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n ;$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{k}}_n .$$

Table I.3: Résumé de l'algorithme FRLS ($7L$)

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 cette constante. En pratique, il faut assurer le démarrage de l'algorithme on pourra choisir par exemple la constante E_0 qui vérifie l'inégalité suivante [36, 39, 40]:

$$E_0 \geq \frac{L}{100} \sigma_x^2 \quad (\text{I.67})$$

où σ_x^2 est l'énergie du signal

I.4 Conclusion

Nous avons posé le problème de l'annulation d'écho acoustique et sa nécessité dans les communications avec les postes mains-libres. En effet, le problème de l'écho acoustique est classiquement résolu à l'aide d'un annulateur d'écho en identifiant progressivement la réponse impulsionnelle finie d'un filtre adaptatif à partir des algorithmes de type moindres carrés. Pour cela, nous avons développé les équations issues de la structure transversale. L'identification du système commence par l'établissement de l'erreur entre la sortie du système et du filtre adaptatif. Par la suite, le choix d'un critère d'erreur nous permet de définir une fonction de coût que nous avons minimisé. Nous avons déduit alors l'équation de Wiener qui répond au problème de minimisation du critère d'erreur choisi. Nous avons rappelé les principaux résultats du LMS et ses versions. Après cela, nous avons développé pour le RLS, les équations issues du Lemme d'inversion de la matrice et son initialisation. Ensuite, nous nous sommes attardés aux algorithmes des moindres carrés récursifs rapides (FRLS). Nous avons vu que la complexité de calcul du FTF augmente linéairement avec l'ordre du filtre adaptatif (L) comme dans le cas de l'algorithme LMS. Ainsi, l'algorithme FTF offre le meilleur des deux mondes: la propriété de convergence de l'algorithme RLS et la simplicité de calcul de l'algorithme LMS.

Malheureusement l'algorithme FTF est exponentiellement instable. Une certaine redondance nous permet de combattre et d'avoir un contrôle sur l'erreur générée par l'accumulation et la propagation des erreurs numériques, qui fera l'objet du prochain chapitre.

Chapitre II

STABILITE NUMERIQUE DES ALGORITHMES FRLS

II.1 Introduction

L'algorithme RLS, basé sur les moindres carrés, possède la caractéristique très intéressante de converger plus rapidement que le gradient stochastique (LMS). Cette convergence rapide se fait au détriment de la complexité de calcul. Le FRLS, version rapide du RLS, basé également sur les moindres carrés, allie la rapidité de convergence du RLS et une complexité de calcul réduite proche du LMS. Cependant, une sérieuse limitation du FRLS est causée par un problème d'instabilité numérique. Une certaine redondance nous permet d'avoir un contrôle sur l'erreur générée par la troncature numérique, qui fera l'objet de ce chapitre.

II.2 Divergence des algorithmes FRLS

Les algorithmes des moindres carrés récursifs rapides (FRLS) sont bien connus pour leur comportement instable et leurs mauvaises propriétés numériques lorsque le facteur d'oubli λ est strictement inférieur à 1. La divergence concernant uniquement la partie prédiction ; la partie filtrage est robuste par rapport à l'implantation numérique, elle nécessite seulement un facteur d'oubli inférieur à 1 [35] et le bon fonctionnement du gain d'adaptation (gain de Kalman) fourni par la partie prédiction. C'est-à-dire, si la partie prédiction est stable alors la partie filtrage l'est aussi. En général, la divergence est détectée quand la variable de vraisemblance devient supérieure à 1 ou bien très faible (tendance vers zéro ou négative), ce qui engendre, après quelques itérations la divergence des principales variables récursives. C'est ce mode de divergence qu'on appelle instabilité numérique, il est plus lié à la structure de l'algorithme qu'à la nature du signal d'entrée. Les erreurs numériques augmentent, au cours du temps, de manière régulière et non bornée. Le meilleur test est le bruit blanc qui permet d'écarter le problème, commun à tous les algorithmes des moindres carrés exacts, dû au mauvais conditionnement de la matrice d'autocorrélation du signal d'entrée. Lorsqu'une précision infinie est utilisée, les

différentes solutions de calcul sont identiques et stables. Même si le problème numérique lié aux nombres finis des calculateurs avait déjà été soulevé dans [41], il fut reporté quelques années plus tard par d'autres auteurs [35, 42]. En effet, J.M. Cioffi proposa de calculer le gain de Kalman avec l'algorithme FRLS (FTF-7L) qui est numériquement instable puis de le réinitialiser à l'aide des paramètres du filtre transversal adaptatif chaque fois que la divergence est détectée. Mais une telle réinitialisation dégrade les performances de l'algorithme. Une réinitialisation périodique plus complexe est proposée dans [43], où le vecteur d'entrée dans la partie prédiction est réinitialisé alors que celui de la partie filtrage est conservé. Dans [44], les auteurs proposent d'amortir périodiquement les filtres de prédiction et régularisent la variance de l'erreur de prédiction aller. Dans [45], l'auteur qui minimise en plus du critère RLS, un critère quadratique incorporant un indicateur de convergence et les filtres prédictifs dans le but de rapprocher les valeurs des filtres de prédiction de leurs valeurs théoriques. Par la suite, les auteurs dans [46-50] introduisent une certaine redondance dans le calcul qui résout partiellement le problème de stabilisation. Nous pensons que la redondance reste la clé du problème du FRLS et il faut introduire une boucle de retour de l'erreur pour résoudre complètement le problème.

II.3 Stabilité numérique des algorithmes FRLS

Le principe de base de la méthode de stabilisation, est de calculer les équations de propagation des erreurs dans les algorithmes FRLS puis de modifier, par une contre réaction convenable, les propriétés numériques de l'algorithme ou plus précisément certaines de ces variables récursives dont l'instabilité numérique est connue ou établie [51]. Cette opération est rendue possible grâce à l'existence de variables redondantes. Une étude complète des erreurs d'arrondi de l'algorithme adaptatif passe par l'étude de la propagation des erreurs [36], les propriétés du bruit d'arrondi et l'accumulation des erreurs. En d'autres termes, comment les erreurs générées à un instant se propagent dans les récursions suivantes, description de l'erreur quand elle est générée à l'instant n et comment les erreurs à différents instants interagissent et s'accumulent.

II.3.1 Approche et Hypothèses

Tout algorithme adaptatif peut être vu comme un système dynamique non linéaire à temps discret, qui s'écrit théoriquement sous la forme d'état suivante:

$$\boldsymbol{\varphi}_n = f[\boldsymbol{\varphi}_{n-1}, \mathbf{x}_n] \quad (\text{II.1})$$

où la fonction $f[\cdot]$ dépend de la structure de l'algorithme. Le vecteur $\boldsymbol{\varphi}_n$ regroupe l'ensemble des variables récursives scalaires et vectorielles de l'algorithme FRLS correspondant à l'analyse, par prédiction linéaire aller et retour d'ordre L , d'un signal x_n .

En précision finie, le système (II.1) s'écrit :

$$\hat{\boldsymbol{\varphi}}_n = f[\hat{\boldsymbol{\varphi}}_{n-1}, \mathbf{x}_n] + \mathbf{e}(n) \quad (\text{II.2})$$

où $\hat{\boldsymbol{\varphi}}_n$ désigne la variable vectorielle calculée par l'algorithme en présence des erreurs numérique représentées par le vecteur $\mathbf{e}(n)$, appelée aussi bruit d'arrondi.

L'erreur introduite sera notée $\Delta\boldsymbol{\varphi}_n$:

$$\Delta\boldsymbol{\varphi}_n = \hat{\boldsymbol{\varphi}}_n - \boldsymbol{\varphi}_n \quad (\text{II.3})$$

L'analyse de la propagation des erreurs numériques dans le système (II.1) est basée sur les hypothèses suivantes :

- Hypothèse H₁: L'analyse se fait au voisinage de la solution théorique de l'algorithme.

Cette hypothèse permet d'obtenir un système de propagation linéaire dépendant du temps :

$$\Delta\boldsymbol{\varphi}_n = \mathbf{F}(n)\Delta\boldsymbol{\varphi}_{n-1} + \mathbf{e}(n) \quad (\text{II.4})$$

où $\mathbf{F}(n)$ représente la matrice de transition des erreurs de propagation, exprimée par :

$$\mathbf{F}(n) = \left. \frac{\partial f[\boldsymbol{\varphi}, \mathbf{x}_n]}{\partial \boldsymbol{\varphi}} \right|_{\boldsymbol{\varphi}=\boldsymbol{\varphi}_{n-1}} \quad (\text{II.5})$$

- Hypothèse H₂: Propagation des erreurs avec une précision infinie.

Ceci permet de réduire le système (II.4) au système libre suivant :

$$\Delta\boldsymbol{\varphi}_n = \mathbf{F}(n)\Delta\boldsymbol{\varphi}_{n-1} \quad (\text{II.6})$$

- Hypothèse H₃: Matrice d'autocorrélation inverse existe

Cette hypothèse suppose que le signal d'entrée x_n est stationnaire et qu'il est tel que l'inverse de la matrice d'autocorrélation à court terme existe, en d'autres termes, que la solution des moindres carrés exacts existe quel que soit l'instant n et $\lambda < 1$.

Le vecteur d'état des équations de propagation des erreurs dans les variables récursives s'écrit :

$$\Delta\boldsymbol{\varphi}_n = \left[\Delta\boldsymbol{\varphi}_n^{aT} \quad \Delta\boldsymbol{\varphi}_n^{kT} \quad \Delta\boldsymbol{\varphi}_n^{bT} \right]^T \quad (\text{II.7a})$$

où

$$\Delta\boldsymbol{\varphi}_n^a = \begin{bmatrix} \Delta\mathbf{a}_n \\ \Delta\alpha_n \end{bmatrix}, \quad \Delta\boldsymbol{\varphi}_n^k = \begin{bmatrix} \Delta\tilde{\mathbf{k}}_n \\ \Delta\gamma_n \end{bmatrix}, \quad \Delta\boldsymbol{\varphi}_n^b = \begin{bmatrix} \Delta\mathbf{b}_n \\ \Delta\beta_n \end{bmatrix} \quad (\text{II.7b})$$

désignent respectivement les erreurs cumulées jusqu'à l'instant n dans les variables récursives aller, Kalman et retour. La matrice $\mathbf{F}(n)$ de dimension $(3L+3) \times (3L+3)$:

$$\mathbf{F}(n) = \begin{bmatrix} \mathbf{F}_{11}(n) & \mathbf{F}_{12}(n) & \mathbf{F}_{13}(n) \\ \mathbf{F}_{21}(n) & \mathbf{F}_{22}(n) & \mathbf{F}_{23}(n) \\ \mathbf{F}_{31}(n) & \mathbf{F}_{32}(n) & \mathbf{F}_{33}(n) \end{bmatrix} \quad (\text{II.8})$$

représente les erreurs de propagation.

Il ne faut pas oublier que l'on désire analyser la propagation des erreurs au voisinage de la solution théorique. Pour l'étude de la propagation des erreurs numériques, le terme supplémentaire (erreurs d'arrondi instantanées $\mathbf{e}(n)$) n'est pas nécessaire ; il est important surtout pour évaluer les effets de la précision finie. Par conséquent, on suppose que les équations de l'algorithme se propagent avec une précision infinie (c'est l'hypothèse H_2) et on étudie l'effet, dans le temps, d'une erreur déterministe $\Delta \boldsymbol{\varphi}_{n_0}$ apparue à l'instant n_0 . C'est la façon dont cette erreur est propagée dans le futur qui détermine la stabilité ou l'instabilité de l'algorithme. Le système d'état (II.4) débarrassé de la perturbation s'écrit alors :

$$\Delta \boldsymbol{\varphi}_n = \mathbf{F}(n) \Delta \boldsymbol{\varphi}_{n-1} \quad (\text{II.9})$$

L'étude de la stabilité du système (II.9) revient à calculer et à étudier les propriétés de la matrice $\mathbf{F}(n)$. Cette matrice sera dite stable si toutes ses valeurs propres sont de module inférieur à 1. On dira aussi, de manière équivalente, qu'un algorithme est stable numériquement si l'effet de n'importe quelle erreur dans (II.9) décroît au cours du temps ; dans le cas contraire, l'algorithme sera dit numériquement instable.

II.3.2 Nouvelle méthode de stabilisation

L'idée consiste à utiliser un mécanisme de retour dans l'ordre d'influence de l'erreur de propagation. Les signaux peuvent revenir en n'importe quel point de l'algorithme sans perturber le vrai caractère de l'algorithme FRLS. Les différences entre une précision infinie et la troncature numérique peuvent être vues comme des signaux de sortie de l'erreur de propagation du système. Nous pouvons penser les utiliser dans un mécanisme de retour dans l'ordre d'influence de l'erreur de propagation. En fait, ces différents signaux peuvent revenir en n'importe quel point de l'algorithme étant donné que la valeur de ces erreurs serait nulle si la précision infinie était disponible. Il existe de multiples manières d'agir pour modifier les propriétés numériques de l'algorithme sans modifier leur structure théorique [50, 51].

Le principe de la méthode de stabilisation est de calculer les équations de propagation des erreurs dans les algorithmes FRLS, puis de modifier par une contre réaction convenable, leurs propriétés numériques.

Ces modifications sont possibles grâce à l'existence de la redondance dans les algorithmes de moindres carrés rapides (FRLS). Il existe différentes manières de calculer l'erreur de prédiction retour a priori. L'une (I.54) est calculée directement sur le signal par la formule de convolution, l'autre (I.63) utilisant dans sa formule la variance des erreurs de prédiction retour β_n et la $(L+1)^{\text{ième}}$ composante du gain de Kalman dual d'ordre $(L+1)$, $\tilde{k}_{L+1,n}^+$. Une nouvelle autre formule qui est la même qu'aux autres est proposée. On montre que la variable de vraisemblance peut se mettre sous la forme suivante [37] :

$$\gamma_n = \lambda^L \frac{\det[\mathbf{R}_{n-1}]}{\det[\mathbf{R}_n]} = \lambda^L \frac{\beta_n}{\alpha_n} \quad (\text{II.10})$$

Cette dernière est obtenue en utilisant les expression (I.37) et (I.65) et les liens entre les déterminants des matrices \mathbf{R}_{n-1} et \mathbf{R}_n . A partir de (II.10), on obtient

$$\beta_n = \lambda^{-L} \gamma_n \alpha_n \quad (\text{II.11})$$

En remplaçant (II.11) à l'instant $(n-1)$ dans (I.63), on abouti facilement à la nouvelle formule utilisant la variable de vraisemblance γ_n , la variance des erreurs de prédiction aller α_n et $\tilde{k}_{L+1,n}^+$..

Les différentes erreurs de prédiction retour a priori sont données par :

$$\bar{r}_n = x_{n-L} - \mathbf{b}_{n-1}^T \mathbf{x}_n \quad (\text{II.12})$$

$$\bar{r}_n^{f_0} = \lambda \beta_{n-1} \tilde{k}_{L+1,n}^+ \quad (\text{II.13})$$

$$\bar{r}_n^{f_1} = \lambda^{-L+1} \gamma_{n-1} \alpha_{n-1} \tilde{k}_{L+1,n}^+ \quad (\text{II.14})$$

En faisant la différence entre ces erreurs de prédiction retour a priori, nous avons défini une variable appelée indicateur de divergence ξ_n [52]:

$$\xi_n = \bar{r}_n - \bar{r}_n^f \quad \begin{cases} = 0 & \text{theorie} \\ \neq 0 & \text{pratique} \end{cases} \quad (\text{II.15})$$

où

$$\bar{r}_n^f = [(1 - \mu_s) \bar{r}_n^{f_0} + \mu_s \bar{r}_n^{f_1}] \quad (\text{II.16})$$

avec

$$0 \leq \mu_s \leq 1 \quad (\text{II.17})$$

En pratique, la variable ξ_n qui dépend des variables $(\mathbf{a}_{n-1}, \alpha_{n-1}; \tilde{\mathbf{k}}_{n-1}, \gamma_{n-1}; \mathbf{b}_{n-1}, \beta_{n-1})$ n'est jamais nulle à cause de la précision des machines utilisées. Cette variable, théoriquement nulle, ne modifie pas la structure de l'algorithme. Par contre, son introduction en un point quelconque de l'algorithme modifie ses propriétés numériques.

Pour stabiliser l'algorithme, nous définissons trois erreurs de prédiction retour a priori, théoriquement équivalentes, qui vont être utilisées pour calculer les variables γ_n, β_n et \mathbf{b}_n :

$$\bar{r}_n^\gamma = \bar{r}_n + \mu^\gamma \xi_n \quad (\text{II.18})$$

$$\bar{r}_n^\beta = \bar{r}_n + \mu^\beta \xi_n \quad (\text{II.19})$$

$$\bar{r}_n^b = \bar{r}_n + \mu^b \xi_n \quad (\text{II.20})$$

où les paramètres scalaires $(\mu^\gamma, \mu^\beta, \mu^b)$ et μ_s permettent de modifier et de contrôler la propagation des erreurs numériques dans l'algorithme ; c'est-à-dire, la modification de la matrice $\mathbf{F}(n)$ afin d'obtenir la stabilité numérique de l'algorithme.

II.3.3 Nouvelles versions d'algorithmes FRLS

Dans ce paragraphe, nous allons présenter de nouvelles versions d'algorithmes numériquement stables (NS-FRLS). Ces versions utilisent des équations qui n'ont pas été exploitées dans la littérature. Ce qui différencie ces algorithmes ce sont les différentes façons de calculer la variable de vraisemblance γ_n . Ils sont donnés par les versions suivantes :

- Version 1 (V_1) :

L'algorithme calcule les variables de vraisemblances d'ordre $L+1$ (γ_n^+) et d'ordre L (γ_n) par les relations suivantes [35]:

$$\gamma_n^+ = \frac{\lambda \alpha_{n-1}}{\alpha_n} \gamma_{n-1} \quad (\text{II.21})$$

$$\gamma_n = \frac{\gamma_n^+}{1 - \gamma_n^+ \bar{r}_n^\gamma \tilde{\mathbf{k}}_{L+1,n}^+} \quad (\text{II.25})$$

Notons que $\mu_s = 0$ de la relation (II.16) correspond à la version (V_B) donnée dans [51].

- Version 2 (V_2) :

Dans cette version, on évite la récursivité dans le calcul de la variable de vraisemblance en utilisant directement la définition:

$$\gamma_n = \frac{1}{1 + \tilde{\mathbf{k}}_n^T \mathbf{x}_n} \quad (\text{II.23})$$

- Version 3 (V₃) :

En reprenant les deux expressions du gain de Kalman dual d'ordre (L+1), calculées au chapitre I:

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} \tilde{\mathbf{k}}_n \\ 0 \end{bmatrix} + \frac{\bar{r}_n}{\lambda\beta_{n-1}} \begin{bmatrix} -\mathbf{b}_{n-1} \\ 1 \end{bmatrix} \quad (\text{II.24})$$

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{n-1} \end{bmatrix} + \frac{\bar{e}_n}{\lambda\alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} \quad (\text{II.25})$$

et en exploitant certaines propriétés d'invariance par décalage du vecteur signal d'entrée étendu à l'ordre (L+1)

$$\mathbf{x}_{L+1,n} = \begin{bmatrix} \mathbf{x}_n^T & x_{n-L} \end{bmatrix}^T \quad (\text{II.26})$$

$$\mathbf{x}_{L+1,n} = \begin{bmatrix} x_n & \mathbf{x}_{n-1}^T \end{bmatrix}^T \quad (\text{II.27})$$

En multipliant à gauche les expressions (II.24) et (II.25) respectivement par (II.26) et (II.27), nous obtenons l'égalité suivante :

$$\mathbf{x}_n^T \tilde{\mathbf{k}}_n + \tilde{k}_{L+1,n}^+ \bar{r}_n = \mathbf{x}_{n-1}^T \tilde{\mathbf{k}}_{n-1} + \tilde{k}_{1,n}^+ \bar{e}_n \quad (\text{II.28})$$

où

$$\tilde{k}_{L+1,n}^+ = \frac{\bar{r}_n}{\lambda\beta_{n-1}} \quad (\text{II.29})$$

$$\tilde{k}_{1,n}^+ = \frac{\bar{e}_n}{\lambda\alpha_{n-1}} \quad (\text{II.30})$$

L'expression (II.28) peut se mettre sous la forme suivante :

$$(1 + \mathbf{x}_n^T \tilde{\mathbf{k}}_n) = (1 + \mathbf{x}_{n-1}^T \tilde{\mathbf{k}}_{n-1}) + \delta_n \quad (\text{II.31})$$

où

$$\delta_n = \tilde{k}_{1,n}^+ \bar{e}_n - \tilde{k}_{L+1,n}^+ \bar{r}_n \quad (\text{II.32})$$

On remarque que le membre de gauche de la relation (II.31) n'est autre que la définition de l'inverse de la variable de vraisemblance à l'instant n . En manipulant cette dernière, on obtient une autre forme de calcul récursive de la variable de vraisemblance :

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \delta_n \gamma_{n-1}} \quad (\text{II.33})$$

- Version 4 (V₄) :

A partir de quelques formules redondantes des algorithmes FRLS, nous pouvons écrire :

$$\frac{\lambda\alpha_{n-1}}{\alpha_n} \gamma_{n-1} = \frac{\lambda\beta_{n-1}}{\beta_n} \gamma_n \quad (\text{II.34})$$

L'équation (II.34) peut être réécrite comme suit :

$$\frac{\alpha_n}{\beta_n} \gamma_n = \frac{\alpha_{n-1}}{\beta_{n-1}} \gamma_{n-1} \quad (\text{II.35})$$

La relation (II.35) que nous notons ψ_n , est une quantité constante à chaque instant n . En utilisant l'expression (II.10) de la variable de vraisemblance γ_n , la relation (II.35) peut être écrite comme suit:

$$\psi_n = \frac{\alpha_n}{\beta_n} \gamma_n = \lambda^L \quad (\text{II.36})$$

En manipulant les équations (II.21) et (II.22), et en utilisant la relation

$$\beta_n = \lambda\beta_{n-1} + \gamma_n (\bar{r}_n^\beta)^2 \quad (\text{II.37})$$

nous obtenons une nouvelle manière de calculer la variable vraisemblance [53]:

$$\gamma_n = \frac{\lambda\alpha_{n-1}}{\alpha_n - \lambda^L (\bar{r}_n^\gamma)^2} \gamma_{n-1} \quad (\text{II.38})$$

Les versions stables, V_{1,2,3} [52] et V₄ [53] des algorithmes des moindres carrés récursifs rapides numériquement stable (NS-FRLS), obtenues sont résumées dans la table II.1. Notons que la version V₁ où les paramètres $\mu^\gamma = \mu^\beta = \mu^b = -1$ et $\mu_s = 0$ correspond à l'algorithme FTF (Fast Transversal Filter) [35], numériquement instable (Fig. II.1).

II.3.4 Résultats de simulation

Nous définissons l'indicateur des erreurs numériques $\sigma_\xi^2(n)$ en dB par la relation suivante :

$$\sigma_\xi^2(n)_{dB} = 10 \log_{10} \left(E \{ \xi_n^2 \} \right) \quad (\text{II.39})$$

$$E \{ \xi_n^2 \} = \frac{1}{M} \sum_{i=1}^M \xi_{i,n}^2 \quad (\text{II.40})$$

où M désigne un certain nombre d'échantillons (256 par exemple). Cette variable représente la variance de l'indicateur de divergence et donne, en décibels, les erreurs numériques qui peuvent être mesurées dans les variables récursives.

Pour un bruit blanc x_n en entrée de moyenne nulle et de variance unité. La figure II.1 représente respectivement l'évolution temporelle de la variable de vraisemblance γ_n et l'indicateur de divergence ξ_n . La figure II.2 représente les erreurs numériques $\sigma_\xi^2(n)$ exprimées en dB. Les paramètres utilisés sont : $L = 32$, $\lambda = 0.99$, $E_0 = 1$ et l'algorithme utilisé est le FTF c'est-à-dire la version V_1 pour le choix ($\mu_s = 0$ et $\mu^\gamma = \mu^\beta = \mu^b = -1$).

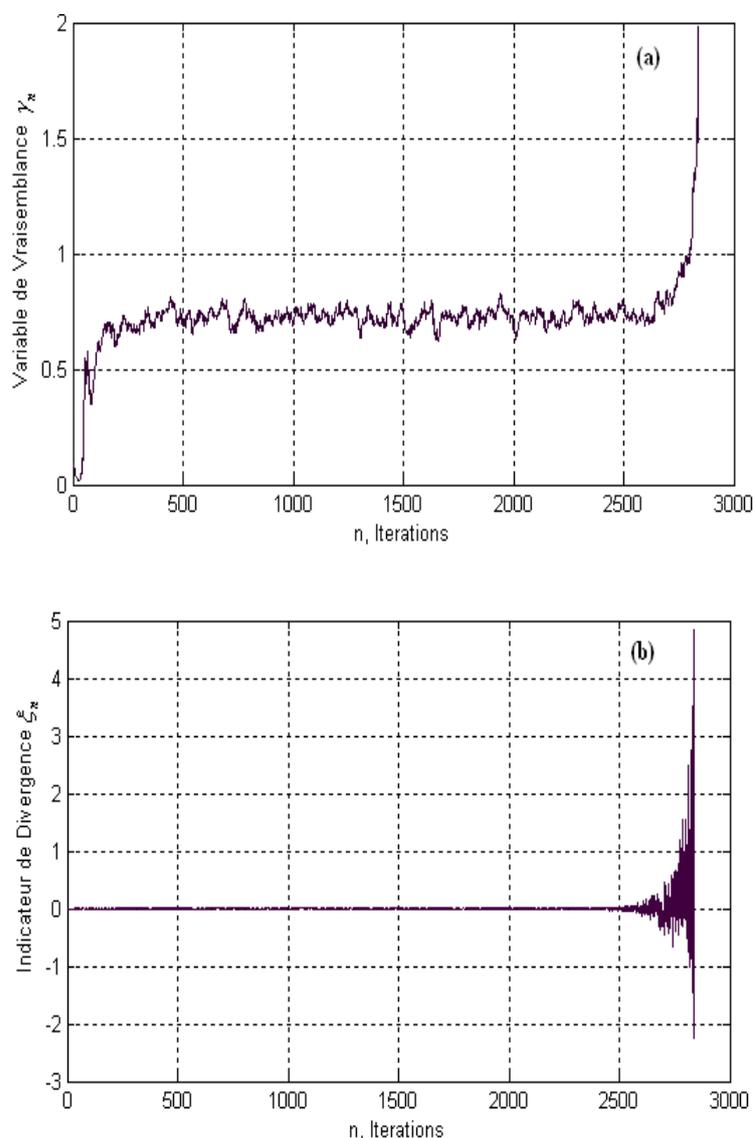


Fig.II.1: Evolution de la variable de Vraisemblance γ_n et de l'Indicateur de Divergence ξ_n

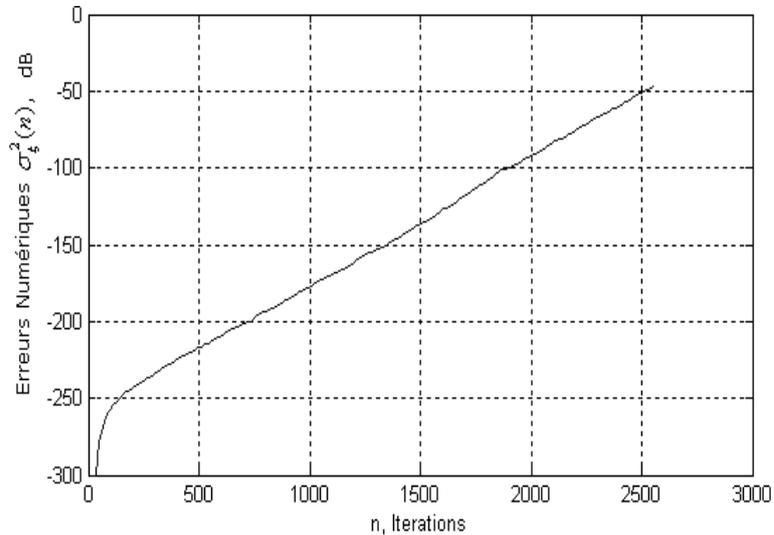


Fig.II.2: Evolution de l'indicateur des erreurs numériques $\sigma_{\xi}^2(n)$

Les simulations montrent, que la divergence est détectée quand la variable de vraisemblance γ_n (Fig.II.1a) devient supérieure à 1. Mais, bien avant cette divergence, on remarque que la variable indicateur de divergence ξ_n (Fig.II.1b) augmente jusqu'à la divergence totale de l'algorithme. L'indicateur des erreurs numériques $\sigma_{\xi}^2(n)$ (Fig.II.2) augmente indéfiniment juste après le démarrage de l'algorithme. Cette courbe croît exponentiellement et arrive au niveau de la variance du signal d'entrée, à ce moment l'algorithme a déjà divergé.

Choix convenable de : $(\mu^\gamma, \mu^\beta, \mu^b)$, μ_s et λ pour un filtre d'ordre L donné

Variables disponibles à l'instant n : $\mathbf{a}_{n-1}; \mathbf{b}_{n-1}; \tilde{\mathbf{k}}_{n-1}; \gamma_{n-1}; \alpha_{n-1}; \beta_{n-1}; \mathbf{w}_{n-1}$

Nouvelles informations: x_n, d_n .

- Partie Prédiction:

Modélisation de: x_n, x_{n-L}

$$\bar{e}_n = x_n - \mathbf{a}_{n-1}^T \mathbf{x}_{n-1};$$

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} \tilde{\mathbf{k}}_n^+ \\ \tilde{k}_{L+1,n}^+ \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{n-1} \end{bmatrix} + \frac{\bar{e}_n}{\lambda \alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix};$$

$$\mathbf{a}_n = \mathbf{a}_{n-1} + \bar{e}_n \gamma_{n-1} \tilde{\mathbf{k}}_{n-1};$$

$$\alpha_n = \lambda \alpha_{n-1} + \gamma_{n-1} \bar{e}_n^2$$

$$\bar{r}_n = x_{n-L} - \mathbf{b}_{n-1}^T \mathbf{x}_n; \quad \bar{r}_n^{f_0} = \lambda \beta_{n-1} \tilde{k}_{L+1,n}^+; \quad \bar{r}_n^{f_1} = \lambda^{-L+1} \gamma_{n-1} \alpha_{n-1} \tilde{k}_{L+1,n}^+;$$

Stabilité Numérique

$$\xi_n = \bar{r}_n - [(1 - \mu_s) \bar{r}_n^{f_0} + \mu_s \bar{r}_n^{f_1}];$$

$$\bar{r}_n^\gamma = \bar{r}_n + \mu^\gamma \xi_n; \quad \bar{r}_n^\beta = \bar{r}_n + \mu^\beta \xi_n; \quad \bar{r}_n^b = \bar{r}_n + \mu^b \xi_n;$$

$$\tilde{\mathbf{k}}_n = \tilde{\mathbf{k}}_n^+ + \tilde{k}_{L+1,n}^+ \mathbf{b}_{n-1};$$

$$\gamma_n^+ = \frac{\lambda \alpha_{n-1}}{\alpha_n} \gamma_{n-1}; \quad \gamma_n = \frac{\gamma_n^+}{1 - \gamma_n^+ \bar{r}_n^\gamma \tilde{k}_{L+1,n}^+} \quad (\text{V}_1);$$

$$\gamma_n = \frac{1}{1 + \tilde{\mathbf{k}}_n^T \mathbf{x}_n} \quad (\text{V}_2);$$

$$\delta_n = \tilde{k}_{1,n}^+ \bar{e}_n - \tilde{k}_{L+1,n}^+ \bar{r}_n^\gamma; \quad \gamma_n = \frac{\gamma_{n-1}}{1 + \delta_n \gamma_{n-1}} \quad (\text{V}_3);$$

$$\gamma_n = \frac{\lambda \alpha_{n-1}}{\alpha_n - \lambda^L (\bar{r}_n^\gamma)^2} \gamma_{n-1} \quad (\text{V}_4);$$

$$\mathbf{b}_n = \mathbf{b}_{n-1} + \bar{r}_n^b \gamma_n \tilde{\mathbf{k}}_n;$$

$$\beta_n = \lambda \beta_{n-1} + \gamma_n (\bar{r}_n^\beta)^2;$$

- Partie Filtrage:

$$\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n;$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{k}}_n.$$

Table II.1: Résumé d'algorithmes NS-FRLS ($V_{1,3,4} : 8L, V_2 : 9L$)

II.4 Propagation des erreurs numériques de la partie prédiction

II.4.1 Calcul des équations de propagation des erreurs dans les variables aller

Les variables récursives de la prédiction aller à analyser sont le prédicteur \mathbf{a}_n et la variance des erreurs de prédiction aller α_n

$$\mathbf{a}_n = \mathbf{a}_{n-1} + \bar{e}_n \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \quad (\text{II.41a})$$

$$\alpha_n = \lambda \alpha_{n-1} + \gamma_{n-1} \bar{e}_n^2 \quad (\text{II.41b})$$

avec

$$\bar{e}_n = x_n - \mathbf{a}_{n-1}^T \mathbf{x}_{n-1} \quad (\text{II.42.})$$

Pour un développement au premier ordre, nous avons:

$$\Delta \mathbf{a}_n = \Delta \mathbf{a}_{n-1} + \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \Delta \bar{e}_n + \bar{e}_n \tilde{\mathbf{k}}_{n-1} \Delta \gamma_{n-1} + \bar{e}_n \gamma_{n-1} \Delta \tilde{\mathbf{k}}_{n-1} \quad (\text{II.43a})$$

$$\Delta \alpha_n = \lambda \Delta \alpha_{n-1} + 2 \gamma_{n-1} \bar{e}_n \Delta \bar{e}_n + \bar{e}_n^2 \Delta \gamma_{n-1} \quad (\text{II.43b})$$

avec

$$\Delta \bar{e}_n = -\mathbf{x}_{n-1}^T \Delta \mathbf{a}_{n-1} \quad (\text{II.44})$$

Sous une autre forme, nous pouvons écrire aussi :

$$\Delta \mathbf{a}_n = \begin{bmatrix} \mathbf{I} - \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T & 0_L \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^a + \bar{e}_n \begin{bmatrix} \gamma_{n-1} \mathbf{I} & \tilde{\mathbf{k}}_{n-1} \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^k \quad (\text{II.45a})$$

$$\Delta \alpha_n = \begin{bmatrix} -2 \bar{e}_n \gamma_{n-1} \mathbf{x}_{n-1}^T & \lambda \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^a + \begin{bmatrix} 0_L^T & \bar{e}_n^2 \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^k \quad (\text{II.45b})$$

où 0_L et \mathbf{I} désignent respectivement un vecteur nul de dimension L et une matrice identité d'ordre L .

En regroupant les expressions (II.45a) et (II.45b), sous une forme plus condensée, il vient :

$$\Delta \boldsymbol{\varphi}_n^a = \begin{bmatrix} \mathbf{I} - \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T & 0_L \\ -2 \bar{e}_n \gamma_{n-1} \mathbf{x}_{n-1}^T & \lambda \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^a + \begin{bmatrix} \bar{e}_n \gamma_{n-1} \mathbf{I} & \bar{e}_n \tilde{\mathbf{k}}_{n-1} \\ 0_L^T & \bar{e}_n^2 \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^k + 0_{(L+1)} \Delta \boldsymbol{\varphi}_{n-1}^b \quad (\text{II.46})$$

Finalement, les erreurs numériques dans le calcul des variables récursives aller ont pour expression :

$$\Delta \boldsymbol{\varphi}_n^a = \mathbf{F}_{11}(n) \Delta \boldsymbol{\varphi}_{n-1}^a + \mathbf{F}_{12}(n) \Delta \boldsymbol{\varphi}_{n-1}^k + \mathbf{F}_{13}(n) \Delta \boldsymbol{\varphi}_{n-1}^b \quad (\text{II.47})$$

où les matrices $\mathbf{F}_{11}(n)$, $\mathbf{F}_{12}(n)$ et $\mathbf{F}_{13}(n)$ se déduisent de la relation (II.46)

II.4.2 Calcul des équations de propagation des erreurs dans les variables de Kalman

Les variables de Kalman à analyser sont le gain de Kalman dual et la variable de vraisemblance d'ordre L des différentes versions:

$$\begin{bmatrix} \tilde{\mathbf{k}}_n \\ 0 \end{bmatrix} = \tilde{\mathbf{k}}_{L+1,n}^+ - \tilde{k}_{L+1,n}^+ \begin{bmatrix} -\mathbf{b}_{n-1} \\ 1 \end{bmatrix} \quad (\text{II.48a})$$

$$\gamma_n = \frac{\gamma_n^+}{1 - \gamma_n^+ \bar{r}_n^\gamma \tilde{k}_{L+1,n}^+} (\text{V}_1); \quad \gamma_n = \frac{1}{1 + \tilde{\mathbf{k}}_n^\top \mathbf{x}_n} (\text{V}_2) \quad (\text{II.48b})$$

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \delta_n \gamma_{n-1}} (\text{V}_3); \quad \gamma_n = \frac{\lambda \alpha_{n-1}}{\alpha_n - \lambda^L (\bar{r}_n^\gamma)^2 \gamma_{L,n-1}} (\text{V}_4)$$

où $\tilde{k}_{L+1,n}^+$ représente la $(L+1)$ ^{ième} composante du gain de Kalman dual d'ordre $(L+1)$ que nous pouvons écrire sous la forme suivante :

$$\tilde{k}_{L+1,n}^+ = \mathbf{q}_{L+1}^\top \tilde{\mathbf{k}}_{L+1,n}^+ \quad (\text{II.49})$$

où le vecteur \mathbf{q}_{L+1} permet d'extraire la $(L+1)$ ^{ième} composante d'un vecteur d'ordre $L+1$

$$\mathbf{q}_{L+1} = [0, 0, \dots, 0, 1]^\top \quad (\text{II.50})$$

L'erreur au premier ordre de l'expression (II.48a) s'écrit :

$$\begin{bmatrix} \Delta \tilde{\mathbf{k}}_n \\ 0 \end{bmatrix} = \left\{ \mathbf{I}_{L+1} - \begin{bmatrix} -\mathbf{b}_{n-1} \\ 1 \end{bmatrix} \mathbf{q}_{L+1}^\top \right\} \Delta \tilde{\mathbf{k}}_{L+1,n}^+ + \tilde{k}_{L+1,n}^+ \begin{bmatrix} \Delta \mathbf{b}_{n-1} \\ 0 \end{bmatrix} \quad (\text{II.51})$$

La relation (II.51) s'écrit sous forme plus condensée :

$$\Delta \tilde{\mathbf{k}}_n = [\mathbf{I} \quad \mathbf{b}_{n-1}] \Delta \tilde{\mathbf{k}}_{L+1,n}^+ + \tilde{k}_{L+1,n}^+ \Delta \mathbf{b}_{n-1} \quad (\text{II.52})$$

La récurrence sur le temps dans le calcul du gain de Kalman dual d'ordre L se fait par l'interaction de ce dernier avec le gain de Kalman dual d'ordre $L+1$, dont l'expression théorique s'écrit :

$$\tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{n-1} \end{bmatrix} + \frac{\bar{e}_n}{\lambda \alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} \quad (\text{II.53})$$

Le calcul d'erreur donne :

$$\Delta \tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} 0 \\ \Delta \tilde{\mathbf{k}}_{n-1} \end{bmatrix} - \frac{\bar{e}_n}{\lambda \alpha_{n-1}^2} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} \Delta \alpha_{n-1} - \frac{1}{\lambda \alpha_{n-1}} \left\{ \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} \mathbf{x}_{n-1}^\top + \bar{e}_n \begin{bmatrix} 0_L^\top \\ \mathbf{I} \end{bmatrix} \right\} \Delta \mathbf{a}_{n-1} \quad (\text{II.54})$$

qui s'écrit sous forme plus condensée :

$$\Delta \tilde{\mathbf{k}}_{L+1,n}^+ = \begin{bmatrix} 0_L^\top \\ \mathbf{I} \end{bmatrix} \Delta \tilde{\mathbf{k}}_{n-1} + \mathbf{Q}_{L+1}(n) \Delta \boldsymbol{\varphi}_{n-1}^a \quad (\text{II.55})$$

où

$$\mathbf{Q}_{L+1}(n) = \frac{1}{\lambda\alpha_{n-1}} \begin{bmatrix} -\mathbf{x}_{n-1}^T & -\frac{\bar{e}_n}{\alpha_{n-1}} \\ \mathbf{a}_{n-1}\mathbf{x}_{n-1}^T - \bar{e}_n\mathbf{I} & \mathbf{a}_{n-1}\frac{\bar{e}_n}{\alpha_{n-1}} \end{bmatrix} \quad (\text{II.56})$$

représente une matrice $(L+1) \times (L+1)$.

En utilisant la relation (II.49), l'erreur dans le calcul de la variable $\tilde{k}_{L+1,n}^+$ s'écrit comme suit :

$$\Delta\tilde{k}_{L+1,n}^+ = \mathbf{q}_{L+1}^T \Delta\tilde{\mathbf{k}}_{L+1,n}^+ = \mathbf{q}_L^T \Delta\tilde{\mathbf{k}}_{n-1} + \mathbf{q}_{L+1}^T \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\varphi}_{n-1}^a \quad (\text{II.57})$$

où

$$\mathbf{q}_{L+1}^T \mathbf{Q}_{L+1}(n) = \frac{1}{\lambda\alpha_{n-1}} \begin{bmatrix} \mathbf{q}_L^T \mathbf{a}_{n-1} \mathbf{x}_{n-1}^T - \bar{e}_n \mathbf{q}_L^T \mathbf{I} & \mathbf{q}_L^T \mathbf{a}_{n-1} \frac{\bar{e}_n}{\alpha_{n-1}} \end{bmatrix} = [\mathbf{v}^T(n) \quad c(n)] \quad (\text{II.58})$$

avec

$$\mathbf{v}^T(n) = \frac{1}{\lambda\alpha_{n-1}} (a_{L,n-1} \mathbf{x}_{n-1}^T - \bar{e}_n \mathbf{q}_L^T) \quad (\text{II.59a}); \quad c(n) = a_{L,n-1} \frac{\bar{e}_n}{\lambda\alpha_{n-1}^2} \quad (\text{II.59b})$$

Sous une forme plus condensée, nous avons :

$$\Delta\tilde{k}_{L+1,n}^+ = [\mathbf{v}^T(n) \quad c(n)] \Delta\boldsymbol{\varphi}_{n-1}^a + [\mathbf{q}_L^T \quad 0] \Delta\boldsymbol{\varphi}_{n-1}^k \quad (\text{II.60})$$

A l'aide des relations (II.52) et (II.55), nous trouvons :

$$\Delta\tilde{\mathbf{k}}_n = \mathbf{M}^c(n) \Delta\tilde{\mathbf{k}}_{n-1} + \mathbf{M}_{L \times (L+1)}^a(n) \Delta\boldsymbol{\varphi}_{n-1}^a + \tilde{k}_{L+1,n}^+ \Delta\mathbf{b}_{n-1} \quad (\text{II.61})$$

avec

$$\mathbf{M}_{L \times (L+1)}^a(n) = [\mathbf{I} \quad \mathbf{b}_{n-1}] \mathbf{Q}_{L+1}(n) \quad (\text{II.62})$$

$$\mathbf{M}^c(n) = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & b_{1,n-1} \\ 1 & 0 & 0 & \cdots & 0 & 0 & b_{2,n-1} \\ 0 & 1 & 0 & \cdots & 0 & 0 & b_{3,n-1} \\ \cdot & & & & & & \\ 0 & 0 & 0 & \cdots & 0 & 1 & b_{L,n-1} \end{bmatrix} \quad (\text{II.63})$$

où $\mathbf{M}_{L \times (L+1)}^a(n)$ représente une matrice $L \times (L+1)$ et $\mathbf{M}^c(n)$ est une matrice compagne d'ordre L dont les valeurs propres sont égales aux pôles instantanés du prédicteur retour.

Enfin sous une forme plus condensée, nous pouvons écrire :

$$\Delta\tilde{\mathbf{k}}_n = \mathbf{M}_{L \times (L+1)}^a(n) \Delta\boldsymbol{\varphi}_{n-1}^a + [\mathbf{M}^c(n) \quad \mathbf{0}_L] \Delta\boldsymbol{\varphi}_{n-1}^k + [\tilde{k}_{L+1,n}^+ \mathbf{I} \quad \mathbf{0}_L] \Delta\boldsymbol{\varphi}_{n-1}^b \quad (\text{II.64})$$

Calculons maintenant les erreurs intervenant dans le calcul de la variable de vraisemblance des différentes versions.

- L'erreur relative dans la variable de vraisemblance de la version 1 (V_1) se calcule comme suit :

$$\gamma_n = \frac{\gamma_n^+}{\theta_n} \Rightarrow \frac{\Delta\gamma_n}{\gamma_n} = \frac{\Delta\gamma_n^+}{\gamma_n^+} - \frac{\Delta\theta_n}{\theta_n} \quad (\text{II.65})$$

$$\gamma_n^+ = \rho_n \gamma_{n-1} \Rightarrow \frac{\Delta\gamma_n^+}{\gamma_n^+} = \frac{\Delta\rho_n}{\rho_n} + \frac{\Delta\gamma_{n-1}}{\gamma_{n-1}} \quad (\text{II.66})$$

avec

$$\rho_n = \frac{\lambda\alpha_{n-1}}{\alpha_n} \Rightarrow \frac{\Delta\rho_n}{\rho_n} = \frac{\Delta\alpha_{n-1}}{\alpha_{n-1}} - \frac{\Delta\alpha_n}{\alpha_n} = (1 - \rho_n) \left\{ \frac{\Delta\alpha_{n-1}}{\alpha_{n-1}} - \frac{\Delta\gamma_{n-1}}{\gamma_{n-1}} - 2 \frac{\Delta\bar{e}_n}{\bar{e}_n} \right\} \quad (\text{II.67})$$

$$\theta_n = 1 - \gamma_n^+ \bar{r}_n^\gamma \tilde{k}_{L+1,n}^+ \Rightarrow \frac{\Delta\theta_n}{\theta_n} = (1 - \theta_n^{-1}) \left\{ \frac{\Delta\gamma_n^+}{\gamma_n^+} + \frac{\Delta\bar{r}_n^\gamma}{\bar{r}_n} + \frac{\Delta\tilde{k}_{L+1,n}^+}{\tilde{k}_{L+1,n}^+} \right\} \quad (\text{II.68})$$

En regroupant les relations (II.66) et (II.68) dans (II.65), il vient :

$$\frac{\Delta\gamma_n}{\gamma_n} = \frac{\rho_n}{\theta_n} \frac{\Delta\gamma_{n-1}}{\gamma_{n-1}} + \frac{(1 - \rho_n)}{\theta_n} \left\{ \frac{\Delta\alpha_{n-1}}{\alpha_{n-1}} - 2 \frac{\Delta\bar{e}_n}{\bar{e}_n} \right\} - (1 - \theta_n^{-1}) \left\{ \frac{\Delta\bar{r}_n^\gamma}{\bar{r}_n} + \frac{\Delta\tilde{k}_{L+1,n}^+}{\tilde{k}_{L+1,n}^+} \right\} \quad (\text{II.69})$$

En utilisant les relations suivantes :

$$(1 - \rho_n) = \frac{\gamma_{n-1}}{\alpha_n} \bar{e}_n^2; \quad (1 - \theta_n^{-1}) = -\gamma_n^+ \bar{r}_n \tilde{k}_{L+1,n}^+; \quad \frac{\gamma_n}{\gamma_{n-1}} = \frac{\rho_n}{\theta_n}; \quad \frac{\gamma_{n-1} \gamma_n}{\theta_n \alpha_n} = \frac{\gamma_n^2}{\lambda \alpha_{n-1}} \quad (\text{II.70})$$

Nous pouvons écrire :

$$\Delta\gamma_n = \frac{2\gamma_n^2 \bar{e}_n}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^T \Delta \mathbf{a}_{n-1} + \frac{(1 - \rho_n) \gamma_n}{\theta_n \alpha_{n-1}} \Delta \alpha_{n-1} + \gamma_n^2 \bar{r}_n \Delta \tilde{k}_{L+1,n}^+ + \gamma_n^2 \tilde{k}_{L+1,n}^+ \Delta \bar{r}_n^\gamma + (\rho_n \theta_n^{-1})^2 \Delta \gamma_{n-1} \quad (\text{II.71})$$

L'erreur dans le calcul de l'erreur de prédiction retour corrigée calculée par la relation

$$\bar{r}_n^\gamma = \bar{r}_n + \mu^\gamma \xi_n \quad (\text{II.72})$$

est

$$\begin{aligned} \Delta \bar{r}_n^\gamma = & -(1 + \mu^\gamma) \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} - \mu^\gamma (1 - \mu_s) \lambda \tilde{k}_{L+1,n}^+ \Delta \beta_{n-1} - \mu^\gamma \lambda \beta_{n-1} \Delta \tilde{k}_{L+1,n}^+ \\ & - \lambda^{-L+1} \mu_s \mu^\gamma \gamma_{n-1} \tilde{k}_{L+1,n}^+ \Delta \alpha_{n-1} - \lambda^{-L+1} \mu_s \mu^\gamma \alpha_{n-1} \tilde{k}_{L+1,n}^+ \Delta \gamma_{n-1} \end{aligned} \quad (\text{II.73})$$

En remplaçant $\Delta \tilde{k}_{L+1,n}^+$ par son expression (II.60), et sous une forme plus condensée nous obtenons:

$$\begin{aligned}
 \Delta \bar{r}_n^\gamma &= \left[-\mu^\gamma \lambda \beta_{n-1} \mathbf{v}^T(n) \quad -\mu^\gamma \lambda \beta_{n-1} c(n) - \lambda^{-L+1} \mu_s \mu^\gamma \gamma_{n-1} \tilde{k}_{L+1,n}^+ \right] \Delta \Phi_{n-1}^a \\
 &+ \left[-\mu^\gamma \lambda \beta_{n-1} \mathbf{q}_L^T \quad -\lambda^{-L+1} \mu_s \mu^\gamma \alpha_{n-1} \tilde{k}_{L+1,n}^+ \right] \Delta \Phi_{n-1}^k \\
 &+ \left[-(1 + \mu^\gamma) \mathbf{x}_n^T \quad -\mu^\gamma (1 - \mu_s) \lambda \tilde{k}_{L+1,n}^+ \right] \Delta \Phi_{n-1}^b
 \end{aligned} \tag{II.74}$$

il vient pour l'erreur numérique dans le calcul de la variable de vraisemblance

$$\begin{aligned}
 \Delta \gamma_n &= \left[\mathbf{v}^{a_1 T}(n) \quad c^{a_1}(n) \right] \Delta \Phi_{n-1}^a \\
 &+ \left[(1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T \quad (\rho_n \theta_n^{-1})^2 - \mu^\gamma \mu_s \lambda^{-L+1} \alpha_{n-1} (\gamma_n \tilde{k}_{L+1,n}^+)^2 \right] \Delta \Phi_{n-1}^k \\
 &+ \left[-\gamma_n^2 (1 + \mu^\gamma) \tilde{k}_{L+1,n}^+ \mathbf{x}_n^T \quad -\lambda \mu^\gamma (1 - \mu_s) (\gamma_n \tilde{k}_{L+1,n}^+)^2 \right] \Delta \Phi_{n-1}^b
 \end{aligned} \tag{II.75}$$

avec

$$\mathbf{v}^{a_1 T} = \frac{2\gamma_n^2 \bar{e}_n}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^T + \gamma_n^2 \bar{r}_n (1 - \mu^\gamma) \mathbf{v}^T(n) \tag{II.76a}$$

$$c^{a_1}(n) = \frac{(1 - \rho_n) \gamma_n}{\theta_n \alpha_{n-1}} + \gamma_n^2 \bar{r}_n (1 - \mu^\gamma) c(n) - \mu_s \mu^\gamma \lambda^{-L+1} \gamma_{n-1} (\gamma_n \tilde{k}_{L+1,n}^+)^2 \tag{II.76b}$$

où $\mathbf{v}^T(n)$ et $c(n)$ sont données par les expressions (II.59a) et (II.59b)

En regroupant la relation (II.64) avec (II.75), on a :

$$\begin{aligned}
 \Delta \Phi_n^k &= \left[\begin{array}{c} \mathbf{M}_{L \times (L+1)}^a(n) \\ \left(\mathbf{v}^{a_1 T}(n), c^{a_1}(n) \right) \end{array} \right] \Delta \Phi_{n-1}^a \\
 &+ \left[\begin{array}{cc} \mathbf{M}^c(n) & \mathbf{0}_L \\ (1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T & (\rho_n \theta_n^{-1})^2 - \mu^\gamma \mu_s \lambda^{-L+1} \alpha_{n-1} (\gamma_n \tilde{k}_{L+1,n}^+)^2 \end{array} \right] \Delta \Phi_{n-1}^k \\
 &+ \left[\begin{array}{cc} \tilde{k}_{L+1,n}^+ \mathbf{I} & \mathbf{0}_L \\ -\gamma_n^2 (1 + \mu^\gamma) \tilde{k}_{L+1,n}^+ \mathbf{x}_n^T & -\lambda \mu^\gamma (1 - \mu_s) (\gamma_n \tilde{k}_{L+1,n}^+)^2 \end{array} \right] \Delta \Phi_{n-1}^b
 \end{aligned} \tag{II.77}$$

Finalement, les erreurs numériques dans le calcul des variables récursives de Kalman, version 1 (V_1), ont pour expression :

$$\Delta \Phi_n^k = \mathbf{F}_{21}^{V_1}(n) \Delta \Phi_{n-1}^a + \mathbf{F}_{22}^{V_1}(n) \Delta \Phi_{n-1}^k + \mathbf{F}_{23}^{V_1}(n) \Delta \Phi_{n-1}^b \tag{II.78}$$

où les matrices $\mathbf{F}_{21}^{V_1}(n)$, $\mathbf{F}_{22}^{V_1}(n)$ et $\mathbf{F}_{23}^{V_1}(n)$ se déduisent de la relation (II.77).

- L'erreur dans la variable de vraisemblance de la version 2 (V_2) s'écrit :

$$\gamma_n = \frac{1}{1 + \tilde{\mathbf{k}}_n^T \mathbf{x}_n} \quad \Rightarrow \quad \Delta \gamma_n = -\gamma_n^2 \mathbf{x}_n^T \Delta \tilde{\mathbf{k}}_n \tag{II.79}$$

Notons que l'utilisation de l'expression (II.79) nous évitera de propager l'erreur de la variable de vraisemblance dans les équations de propagation des erreurs numériques, en utilisant la relation (II.64) les erreurs numériques dans le calcul des variables récursives de Kalman ont pour expression :

$$\Delta\boldsymbol{\phi}_n^k = \begin{bmatrix} \Delta\tilde{k}_n \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{M}_{L \times (L+1)}^a(n) \\ (0_L^T, 0) \end{bmatrix} \Delta\boldsymbol{\phi}_{n-1}^a + \begin{bmatrix} \mathbf{M}^c(n) & 0_L \\ 0_L^T & 0 \end{bmatrix} \Delta\boldsymbol{\phi}_{n-1}^k + \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ 0_L^T & 0 \end{bmatrix} \Delta\boldsymbol{\phi}_{n-1}^b \quad (\text{II.80})$$

Finalement, les erreurs numériques dans le calcul des variables récursives de Kalman, version 2 (V_2), ont pour expression :

$$\Delta\boldsymbol{\phi}_n^k = \mathbf{F}_{21}^{V_2}(n) \Delta\boldsymbol{\phi}_{n-1}^a + \mathbf{F}_{22}^{V_2}(n) \Delta\boldsymbol{\phi}_{n-1}^k + \mathbf{F}_{23}^{V_2}(n) \Delta\boldsymbol{\phi}_{n-1}^b \quad (\text{II.81})$$

où les matrices $\mathbf{F}_{21}^{V_2}(n)$, $\mathbf{F}_{22}^{V_2}(n)$ et $\mathbf{F}_{23}^{V_2}(n)$ se déduisent de la relation (II.80).

- L'erreur relative dans la variable de vraisemblance de la version 3 (V_3) se calcule comme suit :

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \delta_n \gamma_{n-1}} \Rightarrow \frac{\Delta\gamma_n}{\gamma_n} = \frac{\Delta\gamma_{n-1}}{\gamma_{n-1}} - \frac{\delta_n \Delta\gamma_{n-1} + \gamma_{n-1} \Delta\delta_n}{1 + \delta_n \gamma_{n-1}} \quad (\text{II.82})$$

avec

$$\delta_n = \tilde{k}_{1,n}^+ \bar{e}_n - \tilde{k}_{L+1,n}^+ \bar{r}_n^{\gamma} \Rightarrow \Delta\delta_n = \bar{e}_n \Delta\tilde{k}_{1,n}^+ + \tilde{k}_{1,n}^+ \Delta\bar{e}_n - \bar{r}_n \Delta\tilde{k}_{L+1,n}^+ - \tilde{k}_{L+1,n}^+ \Delta\bar{r}_n^{\gamma} \quad (\text{II.83})$$

où $\tilde{k}_{1,n}^+$ représente la première composante du gain de Kalman dual d'ordre ($L+1$) que nous pouvons écrire sous la forme suivante :

$$\tilde{k}_{1,n}^+ = \bar{\mathbf{q}}_{L+1}^T \tilde{\mathbf{k}}_{L+1,n}^+ \quad (\text{II.84})$$

où le vecteur $\bar{\mathbf{q}}_{L+1}$ permet d'extraire la première composante d'un vecteur d'ordre $L+1$

$$\bar{\mathbf{q}}_{L+1} = [1, 0, \dots, 0]^T \quad (\text{II.85})$$

En utilisant la relation (II.84), l'erreur dans le calcul de la variable $\tilde{k}_{1,n}^+$ s'écrit comme suit :

$$\Delta\tilde{k}_{1,n}^+ = \bar{\mathbf{q}}_{L+1}^T \Delta\tilde{\mathbf{k}}_{L+1,n}^+ = \bar{\mathbf{q}}_{L+1}^T \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\phi}_{n-1}^a \quad (\text{II.86})$$

où

$$\bar{\mathbf{q}}_{L+1}^T \mathbf{Q}_{L+1}(n) = -\frac{1}{\lambda \alpha_{n-1}} \begin{bmatrix} \mathbf{x}_{n-1}^T & \frac{\bar{e}_n}{\alpha_{n-1}} \end{bmatrix} = [\bar{\mathbf{v}}^T(n) \quad \bar{c}(n)] \quad (\text{II.87})$$

avec

$$\bar{\mathbf{v}}^T(n) = -\frac{1}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^T \quad (\text{II.88a}); \quad \bar{c}(n) = -\frac{\bar{e}_n}{\lambda \alpha_{n-1}^2} \quad (\text{II.88b})$$

Sous une forme plus condensée, nous avons :

$$\Delta \tilde{k}_{1,n}^+ = [\bar{\mathbf{v}}^T(n) \quad \bar{c}(n)] \Delta \Phi_{n-1}^a \quad (\text{II.89})$$

Nous pouvons écrire :

$$\Delta \gamma_n = \frac{\gamma_n}{\gamma_{n-1}} (1 - \delta_n \gamma_n) \Delta \gamma_{n-1} - \gamma_n^2 \Delta \delta_n \quad (\text{II.90})$$

En remplaçant $\Delta \tilde{k}_{L+1,n}^+$, $\Delta \tilde{k}_{1,n}^+$, $\Delta \bar{e}_n$ et $\Delta \bar{r}_n^\gamma$ par leurs expressions respectives (II.60), (II.89), (II.44) et (II.74), et en manipulant les expressions de (II.70), il vient pour l'erreur numérique dans le calcul de la variable de vraisemblance :

$$\begin{aligned} \Delta \gamma_n &= [\mathbf{v}^{a_3 T}(n) \quad c^{a_3}(n)] \Delta \Phi_{n-1}^a \\ &+ \left[(1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T \quad \rho_n \theta_n^{-1} (\mu^\gamma \mu_s (1 - \theta_n^{-1}) + \rho_n \theta_n^{-1}) \right] \Delta \Phi_{n-1}^k \\ &+ \left[-\gamma_n^2 (1 + \mu^\gamma) \tilde{k}_{L+1,n}^+ \mathbf{x}_n^T \quad -\lambda \mu^\gamma (1 - \mu_s) (\gamma_n \tilde{k}_{L+1,n}^+)^2 \right] \Delta \Phi_{n-1}^b \end{aligned} \quad (\text{II.91})$$

avec

$$\mathbf{v}^{a_3 T} = \gamma_n^2 \bar{r}_n (1 - \mu^\gamma) \mathbf{v}^T(n) - \frac{\gamma_n^2 \bar{e}_n}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^T - \gamma_n^2 \bar{e}_n \bar{\mathbf{v}}^T(n) \quad (\text{II.92a})$$

$$c^{a_3}(n) = \gamma_n^2 \bar{r}_n (1 - \mu^\gamma) c(n) - \mu_s \mu^\gamma \lambda^{-L+1} \gamma_{n-1} (\gamma_n \tilde{k}_{L+1,n}^+)^2 - \gamma_n^2 \bar{e}_n \bar{c}(n) \quad (\text{II.92b})$$

où $(\mathbf{v}^T(n), c(n))$ et $(\bar{\mathbf{v}}^T(n), \bar{c}(n))$ sont données par les expressions (II.59) et (II.88)

En regroupant la relation (II.64) avec (II.91), on a :

$$\begin{aligned} \Delta \Phi_n^k &= \left[\begin{array}{c} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{v}^{a_3 T}(n), c^{a_3}(n)) \end{array} \right] \Delta \Phi_{n-1}^a \\ &+ \left[\begin{array}{cc} \mathbf{M}^c(n) & \mathbf{0}_L \\ (1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T & \rho_n \theta_n^{-1} (\mu^\gamma \mu_s (1 - \theta_n^{-1}) + \rho_n \theta_n^{-1}) \end{array} \right] \Delta \Phi_{n-1}^k \\ &+ \left[\begin{array}{cc} \tilde{k}_{L+1,n}^+ \mathbf{I} & \mathbf{0}_L \\ -\gamma_n^2 (1 + \mu^\gamma) \tilde{k}_{L+1,n}^+ \mathbf{x}_n^T & -\lambda \mu^\gamma (1 - \mu_s) (\gamma_n \tilde{k}_{L+1,n}^+)^2 \end{array} \right] \Delta \Phi_{n-1}^b \end{aligned} \quad (\text{II.93})$$

où $\mathbf{M}_{L \times (L+1)}^a(n)$ et $\mathbf{M}^c(n)$ sont données par les expressions respectives (II.62) et (II.63).

Finalement, les erreurs numériques dans le calcul des variables récursives de Kalman, version 3 (\mathbf{V}_3), ont pour expression :

$$\Delta \Phi_n^k = \mathbf{F}_{21}^{V_3}(n) \Delta \Phi_{n-1}^a + \mathbf{F}_{22}^{V_3}(n) \Delta \Phi_{n-1}^k + \mathbf{F}_{23}^{V_3}(n) \Delta \Phi_{n-1}^b \quad (\text{II.94})$$

où les matrices $\mathbf{F}_{21}^{V_3}(n)$, $\mathbf{F}_{22}^{V_3}(n)$ et $\mathbf{F}_{23}^{V_3}(n)$ se déduisent de la relation (II.93)

- L'erreur relative dans la variable de vraisemblance de la version 4 (V₄) se calcule comme suit :

$$\gamma_n = \frac{\lambda \alpha_{n-1}}{\alpha_n - \lambda^L (\bar{r}_n^\gamma)^2} \gamma_{L,n-1} \Rightarrow \frac{\Delta \gamma_n}{\gamma_n} = \frac{\Delta \alpha_{n-1}}{\alpha_{n-1}} + \frac{\Delta \gamma_{n-1}}{\gamma_{n-1}} - \frac{\Delta \alpha_n - 2\lambda^L \bar{r}_n \Delta \bar{r}_n^\gamma}{\alpha_n - \lambda^L (\bar{r}_n^\gamma)^2} \quad (\text{II.95})$$

En remplaçant $\Delta \alpha_n$ par son expression (II.45b), on aura :

$$\begin{aligned} \Delta \gamma_n = & \frac{2\gamma_n^2 \bar{e}_n}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^\top \Delta \mathbf{a}_{n-1} + \frac{(1 - \rho_n \theta_n^{-1}) \gamma_n}{\alpha_{n-1}} \Delta \alpha_{n-1} + (\rho_n \theta_n^{-1} (1 - \theta_n^{-1}) + (\rho_n \theta_n^{-1})^2) \Delta \gamma_{n-1} \\ & + \frac{2\lambda^{L-1} \bar{r}_n \rho_n \theta_n^{-1} \gamma_n}{\alpha_{n-1}} \Delta \bar{r}_n^\gamma \end{aligned} \quad (\text{II.96})$$

il vient pour l'erreur numérique dans le calcul de la variable de vraisemblance :

$$\begin{aligned} \Delta \gamma_n = & [\mathbf{v}^{a_4^\top}(n) \quad c^{a_4}(n)] \Delta \Phi_{n-1}^a \\ & + \left[-2\mu^\gamma \gamma_{n-1} \rho_n \theta_n^{-1} \bar{r}_n \mathbf{q}_L^\top \quad (1 + 2\mu^\gamma \mu_s) \rho_n \theta_n^{-1} (1 - \theta_n^{-1}) + (\rho_n \theta_n^{-1})^2 \right] \Delta \Phi_{n-1}^k \\ & + \left[-2(1 + \mu^\gamma) \lambda \gamma_n^2 \tilde{k}_{L+1,n}^+ \mathbf{x}_n^\top \quad -2\lambda \mu^\gamma (1 - \mu_s) \lambda (\gamma_n \tilde{k}_{L+1,n}^+)^2 \right] \Delta \Phi_{n-1}^b \end{aligned} \quad (\text{II.97})$$

où

$$\mathbf{v}^{a_4^\top} = \frac{2\gamma_n^2 \bar{e}_n}{\lambda \alpha_{n-1}} \mathbf{x}_{n-1}^\top - 2\mu^\gamma (\gamma_n \bar{r}_n)^2 \mathbf{v}^\top(n) \quad (\text{II.98a})$$

$$c^{a_4}(n) = \frac{\gamma_n}{\alpha_{n-1}} (1 - \rho_n \theta_n^{-1} + 2\mu_s \mu^\gamma (1 - \theta_n^{-1})) - 2\mu^\gamma \gamma_n^2 \bar{r}_n c(n) \quad (\text{II.98b})$$

En regroupant la relation (II.64) avec (II.97), on a :

$$\begin{aligned} \Delta \Phi_n^k = & \left[\begin{array}{c} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{v}^{a_4^\top}(n) \quad c^{a_4}(n)) \end{array} \right] \Delta \Phi_{n-1}^a \\ & + \left[\begin{array}{cc} \mathbf{M}^c(n) & 0_L \\ -2\mu^\gamma \gamma_{n-1} \rho_n \theta_n^{-1} \bar{r}_n \mathbf{q}_L^\top & (1 + 2\mu^\gamma \mu_s) \rho_n \theta_n^{-1} (1 - \theta_n^{-1}) + (\rho_n \theta_n^{-1})^2 \end{array} \right] \Delta \Phi_{n-1}^k \\ & + \left[\begin{array}{cc} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ -2(1 + \mu^\gamma) \lambda \gamma_n^2 \tilde{k}_{L+1,n}^+ \mathbf{x}_n^\top & -2\lambda \mu^\gamma (1 - \mu_s) (\gamma_n \tilde{k}_{L+1,n}^+)^2 \end{array} \right] \Delta \Phi_{n-1}^b \end{aligned} \quad (\text{II.99})$$

Finalement, les erreurs numériques dans le calcul des variables récursives de Kalman, version 4 (V₄), ont pour expression :

$$\Delta \Phi_n^k = \mathbf{F}_{21}^{V_4}(n) \Delta \Phi_{n-1}^a + \mathbf{F}_{22}^{V_4}(n) \Delta \Phi_{n-1}^k + \mathbf{F}_{23}^{V_4}(n) \Delta \Phi_{n-1}^b \quad (\text{II.100})$$

où les matrices $\mathbf{F}_{21}^{V_4}(n)$, $\mathbf{F}_{22}^{V_4}(n)$ et $\mathbf{F}_{23}^{V_4}(n)$ se déduisent de la relation (II.99).

II.4.3 Calcul des équations de propagation des erreurs dans les variables retour

Les variables de la prédiction retour récursives à analyser sont le prédicteur retour et la variance des erreurs de prédictions retour, qui s'écrivent respectivement :

$$\mathbf{b}_n = b_{n-1} + \bar{r}_n^b \gamma_n \tilde{\mathbf{k}}_n \quad (\text{II.101a})$$

$$\beta_n = \lambda \beta_{n-1} + \gamma_n (\bar{r}_n^b)^2 \quad (\text{II.101b})$$

où

$$\bar{r}_n^b = \bar{r}_n + \mu^b \xi_n \quad (\text{II.102a}) ; \quad \bar{r}_n^\beta = \bar{r}_n + \mu^\beta \xi_n \quad (\text{II.102b})$$

représentent les erreurs de prédiction retour corrigées, pour un développement au premier ordre, nous pouvons écrire :

$$\begin{aligned} \Delta \bar{r}_n^b = & -(1 + \mu^b) \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} - \mu^b (1 - \mu_s) \lambda \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \beta_{n-1} - \mu^b \lambda \beta_{n-1} \Delta \tilde{\mathbf{k}}_{L+1,n}^+ \\ & - \lambda^{-L+1} \mu_s \mu^b \gamma_{n-1} \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \alpha_{n-1} - \lambda^{-L+1} \mu_s \mu^b \alpha_{n-1} \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \gamma_{n-1} \end{aligned} \quad (\text{II.103a})$$

$$\begin{aligned} \Delta \bar{r}_n^\beta = & -(1 + \mu^\beta) \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} - \mu^\beta (1 - \mu_s) \lambda \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \beta_{n-1} - \mu^\beta \lambda \beta_{n-1} \Delta \tilde{\mathbf{k}}_{L+1,n}^+ \\ & - \lambda^{-L+1} \mu_s \mu^\beta \gamma_{n-1} \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \alpha_{n-1} - \lambda^{-L+1} \mu_s \mu^\beta \alpha_{n-1} \tilde{\mathbf{k}}_{L+1,n}^+ \Delta \gamma_{n-1} \end{aligned} \quad (\text{II.103b})$$

Les erreurs numériques dans le calcul des expressions (II.101a) et (II.101b) sont respectivement :

$$\Delta \mathbf{b}_n = \Delta \mathbf{b}_{n-1} + \bar{r}_n^b \gamma_n \Delta \tilde{\mathbf{k}}_n + \bar{r}_n^b \tilde{\mathbf{k}}_n \Delta \gamma_n + \gamma_n \tilde{\mathbf{k}}_n \Delta \bar{r}_n^b \quad (\text{II.104a})$$

$$\Delta \beta_n = \lambda \Delta \beta_{n-1} + 2 \bar{r}_n^\beta \gamma_n \Delta \bar{r}_n^\beta + (\bar{r}_n^\beta)^2 \Delta \gamma_n \quad (\text{II.104b})$$

Pour les versions (1, 3 et 4), nous pouvons les mettre sous la forme suivante:

$$\Delta \mathbf{b}_n = \Delta \mathbf{b}_{n-1} + \gamma_n \tilde{\mathbf{k}}_n \Delta \bar{r}_n^b + \begin{bmatrix} \gamma_n \bar{r}_n \mathbf{I} & \bar{r}_n \tilde{\mathbf{k}}_n \end{bmatrix} \Delta \boldsymbol{\varphi}_n^k \quad (\text{II.105a})$$

$$\Delta \beta_n = \lambda \Delta \beta_{n-1} + 2 \bar{r}_n^\beta \gamma_n \Delta \bar{r}_n^\beta + \begin{bmatrix} 0_L^T & \bar{r}_n^2 \end{bmatrix} \Delta \boldsymbol{\varphi}_n^k \quad (\text{II.105b})$$

Tandis que pour la version 2, on aura une autre forme :

$$\Delta \mathbf{b}_n = \Delta \mathbf{b}_{n-1} + \gamma_n \tilde{\mathbf{k}}_n \Delta \bar{r}_n^b + \begin{bmatrix} \gamma_n \bar{r}_n (\mathbf{I} - \gamma_n \tilde{\mathbf{k}}_n \mathbf{x}_n^T) & 0_L \end{bmatrix} \Delta \boldsymbol{\varphi}_n^k \quad (\text{II.106a})$$

$$\Delta \beta_n = \lambda \Delta \beta_{n-1} + 2 \bar{r}_n^\beta \gamma_n \Delta \bar{r}_n^\beta + \begin{bmatrix} -(\bar{r}_n \gamma_n)^2 \mathbf{x}_n^T & 0 \end{bmatrix} \Delta \boldsymbol{\varphi}_n^k \quad (\text{II.106b})$$

En remplaçant $\Delta \bar{r}_n^b$ et $\Delta \bar{r}_n^\beta$ par leurs expressions respectives (II.103a) et (II.103b). Le regroupement des expressions $\Delta \mathbf{b}_n$ et $\Delta \beta_n$ en un seul vecteur de dimension $(L+1)$ donne :

$$\begin{aligned} \Delta \boldsymbol{\varphi}_n^b = & \begin{bmatrix} \mathbf{I} - (1 + \mu^b) \gamma_n \tilde{\mathbf{k}}_n \mathbf{x}_n^T & -\mu^b (1 - \mu_s) \lambda \gamma_n \tilde{\mathbf{k}}_{L+1,n}^+ \tilde{\mathbf{k}}_n \\ -2(1 + \mu^\beta) \gamma_n \bar{r}_n \mathbf{x}_n^T & \lambda \{1 + 2\mu^\beta (1 - \mu_s)(1 - \theta_n^{-1})\} \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^b \\ & + \mathbf{v}_{L+1}^r(n) \Delta \tilde{\mathbf{k}}_{L+1,n}^+ + \mathbf{M}_{L+1}^r(n) \Delta \boldsymbol{\varphi}_n^k \end{aligned} \quad (\text{II.107})$$

avec

$$\mathbf{M}_{L+1}^r(n) = \begin{cases} \begin{bmatrix} \gamma_n \bar{r}_n \mathbf{I} & \bar{r}_n \tilde{\mathbf{k}}_n \\ \mathbf{0}_L^T & \bar{r}_n^2 \end{bmatrix}; & \mathbf{V}_1, \mathbf{V}_3, \mathbf{V}_4 \\ \begin{bmatrix} \gamma_n \bar{r}_n (\mathbf{I} - \gamma_n \tilde{\mathbf{k}}_n \mathbf{x}_n^T) & \mathbf{0}_L \\ -(\bar{r}_n \gamma_n)^2 \mathbf{x}_n^T & 0 \end{bmatrix}; & \mathbf{V}_2 \end{cases} \quad (\text{II.108a})$$

$$\mathbf{v}_{L+1}^r(n) = \lambda \beta_{n-1} \gamma_n \begin{bmatrix} -\mu^b \tilde{\mathbf{k}}_n \\ -2\mu^\beta (1 - \mu_s) \bar{r}_n \end{bmatrix} \quad (\text{II.108b})$$

En remplaçant $\Delta \tilde{\mathbf{k}}_{L+1,n}^+$ et $\Delta \boldsymbol{\varphi}_n^k$ par leurs expressions, nous pouvons écrire :

$$\begin{aligned} \Delta \boldsymbol{\varphi}_n^b &= [\mathbf{M}_{L+1}^r(n) \mathbf{F}_{21}(n) + \mathbf{v}_{L+1}^r(n) [\mathbf{v}^T(n) \quad c(n)]] \Delta \boldsymbol{\varphi}_{n-1}^a \\ &\quad + [\mathbf{M}_{L+1}^r(n) \mathbf{F}_{22}(n) + \mathbf{v}_{L+1}^r(n) [\mathbf{q}_L^T \quad 0]] \Delta \boldsymbol{\varphi}_{n-1}^k \\ &\quad + \left[\mathbf{M}_{L+1}^r(n) \mathbf{F}_{23}(n) + \begin{bmatrix} \mathbf{I} - (1 + \mu^b) \gamma_n \tilde{\mathbf{k}}_n \mathbf{x}_n^T & -\mu^b (1 - \mu_s) \lambda \gamma_n \tilde{\mathbf{k}}_{L+1,n}^+ \tilde{\mathbf{k}}_n \\ -2(1 + \mu^\beta) \gamma_n \bar{r}_n \mathbf{x}_n^T & \lambda \{1 + 2\mu^\beta (1 - \mu_s) (1 - \theta_n^{-1})\} \end{bmatrix} \right] \Delta \boldsymbol{\varphi}_{n-1}^b \end{aligned} \quad (\text{II.109})$$

où les matrices $\mathbf{F}_{21}(n)$, $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{23}(n)$ se déduisent des différentes versions (1, 2, 3 et 4) des relations respectives (II.77), (II.80), (II.93) et (II.99).

Finalement, les erreurs numériques dans le calcul des variables récursives retour ont pour expression :

$$\Delta \boldsymbol{\varphi}_n^b = \mathbf{F}_{31}^{V_i}(n) \Delta \boldsymbol{\varphi}_{n-1}^a + \mathbf{F}_{32}^{V_i}(n) \Delta \boldsymbol{\varphi}_{n-1}^k + \mathbf{F}_{33}^{V_i}(n) \Delta \boldsymbol{\varphi}_{n-1}^b \quad (\text{II.110})$$

où les matrices $\mathbf{F}_{31}^{V_i}(n)$, $\mathbf{F}_{32}^{V_i}(n)$ et $\mathbf{F}_{33}^{V_i}(n)$ se déduisent de la relation (II.97) et l'exposant V_i représente les différentes versions.

II.4.4 Résumé

En résumé, les matrices de propagations des erreurs numériques dans les algorithmes FRLS des différentes versions s'écrivent comme suit :

$$\mathbf{F}_{11}(n) = \begin{bmatrix} \mathbf{I} - \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T & \mathbf{0}_L \\ -2\bar{e}_n \gamma_{n-1} \mathbf{x}_{n-1}^T & \lambda \end{bmatrix}; \quad \mathbf{F}_{12}(n) = \begin{bmatrix} \bar{e}_n \gamma_{n-1} \mathbf{I} & \bar{e}_n \tilde{\mathbf{k}}_{n-1} \\ \mathbf{0}_L^T & \bar{e}_n^2 \end{bmatrix}; \quad \mathbf{F}_{13}(n) = \mathbf{0}_{L \times L}$$

$$\mathbf{F}_{21}(n) = \begin{cases} \begin{bmatrix} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{v}^{a_1 T}(n), c^{a_1}(n)) \end{bmatrix}; & \mathbf{V}_1 \\ \begin{bmatrix} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{0}_L^T, 0) \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{v}^{a_3 T}(n), c^{a_3}(n)) \end{bmatrix}; & \mathbf{V}_3 \\ \begin{bmatrix} \mathbf{M}_{L \times (L+1)}^a(n) \\ (\mathbf{v}^{a_4 T}(n), c^{a_4}(n)) \end{bmatrix}; & \mathbf{V}_4 \end{cases}$$

$$\mathbf{F}_{22}(n) = \begin{cases} \begin{bmatrix} \mathbf{M}^c(n) & \mathbf{0}_L \\ (1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T & (\rho_n \theta_n^{-1})^2 + \mu^\gamma \mu_s \rho_n \theta_n^{-1} (1 - \theta_n^{-1}) \end{bmatrix}; & \mathbf{V}_1 \\ \begin{bmatrix} \mathbf{M}^c(n) & \mathbf{0}_L \\ \mathbf{0}_L^T & 0 \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \mathbf{M}^c(n) & \mathbf{0}_L \\ (1 - \mu^\gamma) \gamma_n^2 \bar{r}_n \mathbf{q}_L^T & \rho_n \theta_n^{-1} (\mu^\gamma \mu_s (1 - \theta_n^{-1}) + \rho_n \theta_n^{-1}) \end{bmatrix}; & \mathbf{V}_3 \\ \begin{bmatrix} \mathbf{M}^c(n) & \mathbf{0}_L \\ -2\mu^\gamma \gamma_{n-1} \rho_n \theta_n^{-1} \bar{r}_n \mathbf{q}_L^T & (1 + 2\mu^\gamma \mu_s) \rho_n \theta_n^{-1} (1 - \theta_n^{-1}) + (\rho_n \theta_n^{-1})^2 \end{bmatrix}; & \mathbf{V}_4 \end{cases}$$

$$\mathbf{F}_{23}(n) = \begin{cases} \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ -\gamma_n^2(1+\mu^\gamma)\tilde{k}_{L+1,n}^+ \mathbf{x}_n^T & -\lambda\mu^\gamma(1-\mu_s)(\gamma_n\tilde{k}_{L+1,n}^+)^2 \end{bmatrix}; & \mathbf{V}_1 \\ \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ 0_L^T & 0 \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ -\gamma_n^2(1+\mu^\gamma)\tilde{k}_{L+1,n}^+ \mathbf{x}_n^T & -\lambda\mu^\gamma(1-\mu_s)(\gamma_n\tilde{k}_{L+1,n}^+)^2 \end{bmatrix}; & \mathbf{V}_3 \\ \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & 0_L \\ -2(1+\mu^\gamma)\gamma_n^2\tilde{k}_{L+1,n}^+ \mathbf{x}_n^T & -2\lambda\mu^\gamma(1-\mu_s)(\gamma_n\tilde{k}_{L+1,n}^+)^2 \end{bmatrix}; & \mathbf{V}_4 \end{cases}$$

$$\mathbf{F}_{31}^{Vi}(n) = \mathbf{M}_{L+1}^r(n)\mathbf{F}_{21}^{Vi}(n) + \mathbf{v}_{L+1}^r(n)[\mathbf{v}^T(n) \quad c(n)]; \quad \mathbf{V}_i (i=1, 2, 3, 4)$$

$$\mathbf{F}_{32}^{Vi}(n) = \mathbf{M}_{L+1}^r(n)\mathbf{F}_{22}^{Vi}(n) + \mathbf{v}_{L+1}^r(n)[\mathbf{q}_L^T \quad 0]; \quad \mathbf{V}_i (i=1, 2, 3, 4)$$

$$\mathbf{F}_{33}(n) =$$

$$\begin{cases} \begin{bmatrix} \theta_n^{-1}\mathbf{I} - \{(1+\mu^b) - (1-\theta_n^{-1})(1+\mu^\gamma)\}\gamma_n\tilde{\mathbf{k}}_n\mathbf{x}_n^T & \lambda(1-\mu_s)\{(1-\theta_n^{-1})\mu^\gamma - \mu^b\}\gamma_n\tilde{k}_{L+1,n}^+\tilde{\mathbf{k}}_n \\ \{(1-\theta_n^{-1})(1+\mu^\gamma) - 2(1+\mu^\beta)\}\gamma_n\bar{r}_n\mathbf{x}_n^T & \lambda\{1+(1-\mu_s)(1-\theta_n^{-1})(2\mu^\beta - \mu^\gamma(1-\theta_n^{-1}))\} \end{bmatrix}; & \mathbf{V}_1, \mathbf{V}_3 \\ \begin{bmatrix} \theta_n^{-1}\mathbf{I} - (\mu^b + \theta_n^{-1})\gamma_n\tilde{\mathbf{k}}_n\mathbf{x}_n^T & -\mu^b(1-\mu_s)\lambda\gamma_n\tilde{k}_{L+1,n}^+\tilde{\mathbf{k}}_n \\ \{(1-\theta_n^{-1}) - 2(1+\mu^\beta)\}\gamma_n\bar{r}_n\mathbf{x}_n^T & \lambda\{1+2\mu^\beta(1-\mu_s)(1-\theta_n^{-1})\} \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \theta_n^{-1}\mathbf{I} - \{(1+\mu^b) - 2(1-\theta_n^{-1})(1+\mu^\gamma)\}\gamma_n\tilde{\mathbf{k}}_n\mathbf{x}_n^T & \lambda(1-\mu_s)\{2(1-\theta_n^{-1})\mu^\gamma - \mu^b\}\gamma_n\tilde{k}_{L+1,n}^+\tilde{\mathbf{k}}_n \\ 2\{(1-\theta_n^{-1})(1+\mu^\gamma) - 2(1+\mu^\beta)\}\gamma_n\bar{r}_n\mathbf{x}_n^T & \lambda\{1+2(1-\mu_s)(1-\theta_n^{-1})(\mu^\beta - \mu^\gamma(1-\theta_n^{-1}))\} \end{bmatrix}; & \mathbf{V}_4 \end{cases}$$

$\mathbf{Q}_{L+1}(n)$: Matrice donnée par (II.56)

$(\mathbf{v}^T(n), \quad c(n))$: (Vecteur ligne, scalaire) donné par (II.59)

$\mathbf{M}_{L \times (L+1)}^a(n)$: Matrice donnée par (II.62)

$\mathbf{M}^c(n)$: Matrice donnée par (II.63)

$(\mathbf{v}^{a_1^T}(n), \quad c^{a_1}(n))$: (Vecteur ligne, scalaire) donné par (II.76)

$(\mathbf{v}^{a_3^T}(n), \quad c^{a_3}(n))$: (Vecteur ligne, scalaire) donné par (II.88)

$(\mathbf{v}^{a_4^T}(n), \quad c^{a_4}(n))$: (Vecteur ligne, scalaire) donné par (II.98)

$\mathbf{M}_{L+1}^r(n)$: Matrice donnée par (II.108a)

$\mathbf{v}_{L+1}^r(n)$: Vecteur donné par (II.108b)

II.5 Conclusion

Nous avons calculé les équations de propagation des erreurs dans l'algorithme FRLS, puis modifié par une contre réaction convenable, les propriétés numériques de l'algorithme. Nous avons proposé quatre nouvelles versions d'algorithmes numériquement stables. Ce qui différencie ces algorithmes ce sont les différentes façons de calculer la variable de vraisemblance. Ces versions sont rendues possibles grâce à l'existence de variables redondantes. Nous avons donné les matrices de propagations des erreurs numériques dans les algorithmes FRLS modifiés. Ces nouvelles méthodes augmentent faiblement la complexité des algorithmes initiaux de L multiplications supplémentaire pour chaque méthode.

Chapitre III

ANALYSE DES ERREURS NUMERIQUES DES ALGORITHMES FRLS

III.1 Introduction

Les algorithmes des moindres carrés rapides FRLS sont bien connus pour leurs comportements instables et leurs mauvaises propriétés numériques lorsque le facteur d'oubli est strictement inférieur à 1. Les erreurs numériques se propagent d'une manière non bornée au cours du temps, ce qui conduit à une solution instable plus au moins long terme. Nous allons présenter l'analyse de la propagation des erreurs numériques de la partie prédiction des algorithmes FRLS modifiés par leurs nouvelles définitions que nous appelons algorithmes des moindres carrés récurifs rapides numériquement stables (NS-FRLS). Après la modification de l'algorithme et le calcul des nouvelles équations de propagation des erreurs numériques, nous allons étudier comment éviter l'instabilité numérique. Nous montrons, pour un signal stationnaire en entrée, que ces algorithmes ainsi modifiés sont stables numériquement pour un choix convenable des paramètres et du facteur d'oubli. Nous présentons une étude détaillée pour des signaux parfaitement prédictibles, cas des sinusoïdes pures. Nous proposons, afin d'éviter l'instabilité de l'algorithme FRLS, un choix de la fréquence d'échantillonnage par rapport à la fréquence de la sinusoïde.

III.2 Analyse des erreurs numériques de la partie prédiction

III.2.1 Analyse des erreurs numériques

L'étude de la stabilité de la matrice de propagation des erreurs numérique $\mathbf{F}(n)$ dans toute sa généralité est une tâche très difficile à cause de sa complexité. Par conséquent, nous pouvons en déduire pour chaque version que nous avons développée, que la matrice $\mathbf{F}_{13}(n)$ est nulle, et nous pouvons dire aussi que la matrice $\mathbf{F}_{12}(n)$ peut être rendue négligeable par le choix d'un facteur d'oubli λ très proche de 1. La matrice $\mathbf{F}_{23}(n)$ donnée par :

$$\mathbf{F}_{23}(n) = \begin{cases} \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & \mathbf{0}_L \\ -\gamma_n^2(1+\mu^\gamma)\tilde{k}_{L+1,n}^+ \mathbf{x}_n^\top & -\lambda\mu^\gamma(1-\mu_s)(\gamma_n\tilde{k}_{L+1,n}^+)^2 \end{bmatrix}; & \mathbf{V}_{1,3} \\ \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & \mathbf{0}_L \\ \mathbf{0}_n^\top & 0 \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \tilde{k}_{L+1,n}^+ \mathbf{I} & \mathbf{0}_L \\ -2(1+\mu^\gamma)\gamma_n^2\tilde{k}_{L+1,n}^+ \mathbf{x}_n^\top & -2\lambda\mu^\gamma(1-\mu_s)(\gamma_n\tilde{k}_{L+1,n}^+)^2 \end{bmatrix}; & \mathbf{V}_4 \end{cases} \quad (\text{III.1})$$

peut être négligeable. En reprenant l'expression suivante :

$$\tilde{k}_{L+1,n}^+ = \mathbf{q}_{L+1}^\top \tilde{\mathbf{k}}_{L+1,n}^+ = \lambda^{-1} \mathbf{q}_{L+1}^\top \mathbf{R}_{L+1,n-1}^{-1} \mathbf{x}_{L+1,n} \quad (\text{III.2})$$

et en utilisant l'approximation suivante :

$$\tilde{k}_{L+1,n}^+ \cong -(1-\lambda^{-1}) \mathbf{q}_{L+1}^\top \mathbf{R}_{L+1,X}^{-1} \mathbf{x}_{L+1,n} \quad (\text{III.3})$$

avec

$$\mathbf{R}_{L+1,X} = \mathbf{E}\{\mathbf{x}_{L+1,n} \mathbf{x}_{L+1,n}^\top\} \quad (\text{III.4})$$

Le coefficient $\tilde{k}_{L+1,n}^+$, peut être rendue aussi faible que l'on veut par le choix d'un facteur d'oubli λ , il suffit que la matrice d'autocorrélation reste bornée. Dans ce cas, il est facile de voir que la matrice $\mathbf{F}_{23}(n)$ a un poids très faible, pour le choix de λ très proche de 1.

Dans ce cas, la matrice $\mathbf{F}(n)$ est très proche d'une matrice triangulaire inférieure par blocs. Ce qui revient d'écrire:

$$\mathbf{F}(n) = \begin{bmatrix} \mathbf{F}_{11}(n) & \mathbf{0}_{L+1} & \mathbf{0}_{L+1} \\ \mathbf{F}_{21}(n) & \mathbf{F}_{22}(n) & \mathbf{0}_{L+1} \\ \mathbf{F}_{31}(n) & \mathbf{F}_{32}(n) & \mathbf{F}_{33}(n) \end{bmatrix} \quad (\text{III.5})$$

Par conséquent, nous pouvons approcher les erreurs cumulées jusqu'à l'instant n dans les variables récursives aller $\Delta\boldsymbol{\varphi}_n^a$, Kalman $\Delta\boldsymbol{\varphi}_n^k$ et retour $\Delta\boldsymbol{\varphi}_n^b$ par :

$$\Delta\boldsymbol{\varphi}_n^a = \mathbf{F}_{11}(n)\Delta\boldsymbol{\varphi}_{n-1}^a \quad (\text{III.6a})$$

$$\Delta\boldsymbol{\varphi}_n^k = \mathbf{F}_{22}(n)\Delta\boldsymbol{\varphi}_{n-1}^k + \mathbf{T}_k [\Delta\boldsymbol{\varphi}_{n-1}^a] \quad (\text{III.6b})$$

$$\Delta\boldsymbol{\varphi}_n^b = \mathbf{F}_{33}(n)\Delta\boldsymbol{\varphi}_{n-1}^b + \mathbf{T}_b [\Delta\boldsymbol{\varphi}_{n-1}^a; \Delta\boldsymbol{\varphi}_{n-1}^k] \quad (\text{III.6c})$$

où $\mathbf{T}_k[\cdot]$ et $\mathbf{T}_b[\cdot]$, fonctions non explicites, qui ne dépendent pas de $\Delta\boldsymbol{\varphi}_{n-1}^k$ et $\Delta\boldsymbol{\varphi}_{n-1}^b$ respectivement. La matrice $\mathbf{F}_{11}(n)$ donnée par :

$$\mathbf{F}_{11}(n) = \begin{bmatrix} \mathbf{I} - \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T & \mathbf{0}_L \\ -2\bar{\epsilon}_n \gamma_{n-1} \mathbf{x}_{n-1}^T & \lambda \end{bmatrix} \quad (\text{III.7})$$

peut être écrite autrement, en développant l'expression suivante :

$$\gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T = \mathbf{R}_{n-1}^{-1} \mathbf{x}_{n-1} \mathbf{x}_{n-1}^T = \mathbf{R}_{n-1}^{-1} (\mathbf{R}_{n-1} - \lambda \mathbf{R}_{n-2}) = \mathbf{I} - \lambda \mathbf{R}_{n-1}^{-1} \mathbf{R}_{n-2} \quad (\text{III.8})$$

En utilisant l'approximation suivante :

$$\mathbf{I} - \gamma_{n-1} \tilde{\mathbf{k}}_{n-1} \mathbf{x}_{n-1}^T \cong \lambda \mathbf{I} \quad (\text{III.9})$$

Nous pouvons dire que la matrice

$$\mathbf{F}_{11}(n) \cong \begin{bmatrix} \lambda \mathbf{I} & \mathbf{0}_L \\ * & \lambda \end{bmatrix} \quad (\text{III.10})$$

a toutes ses valeurs propres inférieures à 1. La quantité (*) de (III.10) n'influence pas sur l'étude.

Les matrices $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$ peuvent s'écrire comme suit :

$$\mathbf{F}_{22}(n) = \begin{bmatrix} \mathbf{M}_k(n) & \mathbf{u}_1(n) \\ \mathbf{u}_2^T(n) & c_\gamma(n) \end{bmatrix} \quad (\text{III.11a})$$

$$\mathbf{F}_{33}(n) = \begin{bmatrix} \mathbf{M}_b(n) & \mathbf{v}_1(n) \\ \mathbf{v}_2^T(n) & c_\beta(n) \end{bmatrix} \quad (\text{III.11b})$$

avec $\mathbf{u}_1(n) = \mathbf{0}_L$ (vecteur nul) et $\mathbf{v}_1(n)$ vecteur donnée par :

$$\mathbf{v}_1(n) = \begin{cases} \lambda(1 - \mu_s) \{ (1 - \theta_n^{-1}) \mu^\gamma - \mu^b \} \gamma_n \tilde{k}_{L+1,n}^+ \tilde{\mathbf{k}}_n; & \mathbf{V}_{1,3} \\ -\mu^b (1 - \mu_s) \lambda \gamma_n \tilde{k}_{L+1,n}^+ \tilde{\mathbf{k}}_n; & \mathbf{V}_2 \\ \lambda(1 - \mu_s) \{ 2(1 - \theta_n^{-1}) \mu^\gamma - \mu^b \} \gamma_n \tilde{k}_{L+1,n}^+ \tilde{\mathbf{k}}_n; & \mathbf{V}_4 \end{cases} \quad (\text{III.12})$$

En utilisant l'approximation (III.3) sur $\tilde{k}_{L+1,n}^+$, un facteur d'oubli λ proche de 1 permet d'affaiblir l'influence du vecteur $\mathbf{v}_1(n)$. Les vecteurs $\mathbf{u}_2(n)$ et $\mathbf{v}_2(n)$ n'influencent pas sur l'étude. Nous pouvons approcher les erreurs numériques dans le calcul de $\Delta \boldsymbol{\varphi}_n^k$ et $\Delta \boldsymbol{\varphi}_n^b$ par des modèles du premier ordre déduites respectivement de $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$:

$$\Delta \tilde{\mathbf{k}}_n = \mathbf{M}_k(n) \Delta \tilde{\mathbf{k}}_{n-1} + \mathbf{p}_k(n) \quad (\text{III.13a})$$

$$\Delta \gamma_n = c_\gamma(n) \Delta \gamma_{n-1} + p_\gamma(n) \quad (\text{III.13b})$$

$$\Delta \mathbf{b}_n = \mathbf{M}_b(n) \Delta \mathbf{b}_{n-1} + \mathbf{p}_b(n) \quad (\text{III.14a})$$

$$\Delta \beta_n = c_\beta(n) \Delta \beta_{n-1} + p_\beta(n) \quad (\text{III.14b})$$

avec $\mathbf{M}_k(n) = \mathbf{M}^c(n)$ représentant la matrice compagnon donnée par (II.63) et

$$c_\gamma(n) = \begin{cases} \mu^\gamma \mu_s (1 - \theta_n^{-1}) \rho_n \theta_n^{-1} + (\rho_n \theta_n^{-1})^2; & \mathbf{V}_{1,3} \\ 0; & \mathbf{V}_2 \\ (1 + 2\mu^\gamma \mu_s)(1 - \theta_n^{-1}) \rho_n \theta_n^{-1} + (\rho_n \theta_n^{-1})^2; & \mathbf{V}_4 \end{cases} \quad (\text{III.15})$$

$$\mathbf{M}_b(n) = [\theta_n^{-1} \mathbf{I} - c_b(n) \gamma_n \tilde{\mathbf{K}}_n \mathbf{x}_n^T] \quad (\text{III.16a})$$

$$c_b(n) = \begin{cases} \{(1 + \mu^b) - (1 - \theta_n^{-1})(1 + \mu^\gamma)\}; & \mathbf{V}_{1,3} \\ (\mu^b + \theta_n^{-1}); & \mathbf{V}_2 \\ \{(1 + \mu^b) - 2(1 - \theta_n^{-1})(1 + \mu^\gamma)\}; & \mathbf{V}_4 \end{cases} \quad (\text{III.16b})$$

$$c_\beta(n) = \begin{cases} \lambda \{1 + (1 - \mu_s)(1 - \theta_n^{-1})(2\mu^\beta - \mu^\gamma(1 - \theta_n^{-1}))\}; & \mathbf{V}_{1,3} \\ \lambda \{1 + 2\mu^\beta(1 - \mu_s)(1 - \theta_n^{-1})\}; & \mathbf{V}_2 \\ \lambda \{1 + 2(1 - \mu_s)(1 - \theta_n^{-1})(\mu^\beta - \mu^\gamma(1 - \theta_n^{-1}))\}; & \mathbf{V}_4 \end{cases} \quad (\text{III.17})$$

$$\rho_n = \frac{\lambda \alpha_{n-1}}{\alpha_n} \quad (\text{III.18a}); \quad \theta_n = \frac{\lambda \beta_{n-1}}{\beta_n} \quad (\text{III.18b})$$

où les termes de perturbations ($\mathbf{p}_k(n)$, $p_\gamma(n)$, $\mathbf{p}_b(n)$ et $p_\beta(n)$) peuvent être facilement trouvés à partir du système (III.6).

III.2.2 Choix des paramètres de contrôle et condition de stabilité

En supposant que les termes de perturbations ($\mathbf{p}_k(n)$, $p_\gamma(n)$, $\mathbf{p}_b(n)$ et $p_\beta(n)$) restent bornés ; le choix des paramètres de contrôle, pour que le système soit stable, revient à étudier les gains scalaires ($c_\gamma(n)$, $c_\beta(n)$) et matriciels ($\mathbf{M}_k(n)$, $\mathbf{M}_b(n)$) ; c'est-à-dire il faut que ces gains soient inférieurs à 1. Il existe une infinité de solution pour le choix des paramètres à partir de ces équations. A chaque solution correspond une condition de stabilité sur λ .

Notons que les variables ρ_n , θ_n et \mathbf{R}_n sont des quantités lentement variables par rapport au signal d'entrée x_n . Les variable ρ_n (III.18a) et θ_n (III.18b) ont pour valeur asymptotique λ et l'expression :

$$\gamma_n \tilde{\mathbf{K}}_n \mathbf{x}_n^T = \mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T = \mathbf{R}_n^{-1} (\mathbf{R}_n - \lambda \mathbf{R}_{n-1}) = \mathbf{I} - \lambda \mathbf{R}_n^{-1} \mathbf{R}_{n-1} \quad (\text{III.19})$$

aura comme valeur asymptotique $(1 - \lambda)\mathbf{I}$.

En régime asymptotique, les matrices \mathbf{F}_{22} et \mathbf{F}_{33} peuvent se mettre comme suit :

$$\mathbf{F}_{22} = \begin{cases} \begin{bmatrix} \mathbf{M}^c & 0_L \\ * & \mu^\gamma \mu_s (1 - \lambda^{-1}) + 1 \end{bmatrix}; & \mathbf{V}_{1,3} \\ \begin{bmatrix} \mathbf{M}^c & 0_L \\ * & 0 \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \mathbf{M}^c & 0_L \\ * & 1 + (1 + 2\mu^\gamma \mu_s)(1 - \lambda^{-1}) \end{bmatrix}; & \mathbf{V}_4 \end{cases} \quad (\text{III.20a})$$

$$\mathbf{F}_{33} = \begin{cases} \begin{bmatrix} \{(1 + 2\mu^\gamma - \mu^b) + (\mu^b - \mu^\gamma)\lambda - \mu^\gamma \lambda^{-1}\} \mathbf{I} & \mathbf{v}_1 \rightarrow 0_L \\ * & 2(\mu^\gamma - \mu^\beta)(1 - \mu_s) + (1 + (1 - \mu_s)(2\mu^\beta - \mu^\gamma))\lambda - \mu^\gamma (1 - \mu_s)\lambda^{-1} \end{bmatrix}; & \mathbf{V}_{1,3} \\ \begin{bmatrix} \{(1 - \mu^b) + \mu^b \lambda\} \mathbf{I} & \mathbf{v}_1 \rightarrow 0_L \\ * & -2\mu^\beta (1 - \mu_s) + (1 + 2\mu^\beta (1 - \mu_s))\lambda \end{bmatrix}; & \mathbf{V}_2 \\ \begin{bmatrix} \{(3 + 4\mu^\gamma - \mu^b) - (1 + 2\mu^\gamma - \mu^b)\lambda - (1 + 2\mu^\gamma)\lambda^{-1}\} \mathbf{I} & \mathbf{v}_1 \rightarrow 0_L \\ * & 2(1 - \mu_s)(2\mu^\gamma - \mu^\beta) + (1 + 2(1 - \mu_s)(\mu^\beta - \mu^\gamma))\lambda - 2(1 - \mu_s)\mu^\gamma \lambda^{-1} \end{bmatrix}; & \mathbf{V}_4 \end{cases} \quad (\text{III.20b})$$

Pour la version \mathbf{V}_1 et avec le choix des paramètres de contrôle ($\mu^\gamma = \mu^\beta = \mu^b = -1$ et $\mu_s = 0$), les matrices \mathbf{F}_{22} et \mathbf{F}_{33} ont toutes leurs valeurs propres supérieures à 1. Ce choix correspond à l'algorithme FTF (Fast Transversal Filter) [35], numériquement instable (Fig. II.1) vu au précédent chapitre. A cet effet, il faut choisir convenablement les paramètres de contrôle pour bien conditionner ces matrices. Les tableaux III.1 à III.4 représentent les matrices \mathbf{F}_{22} et \mathbf{F}_{33} pour les choix respectifs :

$$\mathbf{C}_1: (\mu^\gamma = \mu^\beta = \mu^b = 1);$$

$$\mathbf{C}_2: (\mu^\gamma = 0, \mu^\beta = \mu^b = 1);$$

$$\mathbf{C}_3: (\mu^\gamma = 1, \mu^\beta = 1.5, \mu^b = 0.5);$$

$$\mathbf{C}_4: (\mu^\gamma = 0, \mu^\beta = 1.5, \mu^b = 0.5).$$

versions	\mathbf{F}_{22}		\mathbf{F}_{33}	
V ₁	\mathbf{M}^c *	0_L $\mu_s(1-\lambda^{-1})+1$	$(2-\lambda^{-1})\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+(1-\mu_s)(1-\lambda^{-2})\}$
V ₂	\mathbf{M}^c *	0_L 0	$\lambda\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+2(1-\mu_s)(1-\lambda^{-1})\}$
V ₃	\mathbf{M}^c *	0_L $1+\mu_s(1-\lambda^{-1})$	$(2-\lambda^{-1})\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+(1-\mu_s)(1-\lambda^{-2})\}$
V ₄	\mathbf{M}^c *	0_L $1+(1+2\mu_s)(1-\lambda^{-1})$	$(6-3\lambda^{-1}-2\lambda)\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda+2(1-\mu_s)(1-\lambda^{-1})$

Tableau III.1: Matrices $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$ en régime asymptotique pour le choix C_1 ($\mu^\gamma = \mu^\beta = \mu^b = 1$); $0 \leq \mu_s \leq 1$; La quantité (*) n'influence pas sur l'étude.

versions	\mathbf{F}_{22}		\mathbf{F}_{33}	
V ₁	\mathbf{M}^c *	0_L 1	$\lambda\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+2(1-\mu_s)(1-\lambda^{-1})\}$
V ₂	\mathbf{M}^c *	0_L 0	$\lambda\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+2(1-\mu_s)(1-\lambda^{-1})\}$
V ₃	\mathbf{M}^c *	0_L 1	$\lambda\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+2(1-\mu_s)(1-\lambda^{-1})\}$
V ₄	\mathbf{M}^c *	0_L $(2-\lambda^{-1})$	$(2-\lambda^{-1})\mathbf{I}$ *	$\mathbf{v}_1 \rightarrow 0_L$ $\lambda\{1+2(1-\mu_s)(1-\lambda^{-1})\}$

Tableau III.2: Matrices $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$ en régime asymptotique pour le choix C_2 ($\mu^\gamma = 0$, $\mu^\beta = \mu^b = 1$); $0 \leq \mu_s \leq 1$; La quantité (*) n'influence pas sur l'étude.

versions	\mathbf{F}_{22}		\mathbf{F}_{33}	
V ₁	\mathbf{M}^c	0_L	$(2.5 - 0.5\lambda - \lambda^{-1})\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	$\mu_s(1 - \lambda^{-1}) + 1$	*	$\lambda\{1 + (1 - \mu_s)(2 - \lambda^{-1} - \lambda^{-2})\}$
V ₂	\mathbf{M}^c	0_L	$(2.5 - 0.5\lambda - \lambda^{-1})\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	0	*	$\lambda\{1 + 3(1 - \mu_s)(1 - \lambda^{-1})\}$
V ₃	\mathbf{M}^c	0_L	$(0.5 + 0.5\lambda)\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	$1 + \mu_s(1 - \lambda^{-1})$	*	$\lambda\{1 + (1 - \mu_s)(2 - \lambda^{-1} - \lambda^{-2})\}$
V ₄	\mathbf{M}^c	0_L	$(6.5 - 2.5\lambda - 3\lambda^{-1})\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	$1 + (1 + 2\mu_s)(1 - \lambda^{-1})$	*	$\lambda\{1 + (1 - \mu_s)(1 + \lambda^{-1} - 2\lambda^{-2})\}$

Tableau III.3: Matrices $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$ en régime asymptotique pour le choix

\mathbf{C}_3 ($\mu^\gamma = 1$, $\mu^\beta = 1.5$, $\mu^b = 0.5$); $0 \leq \mu_s \leq 1$; La quantité (*) n'influence pas sur l'étude.

versions	\mathbf{F}_{22}		\mathbf{F}_{33}	
V ₁	\mathbf{M}^c	0_L	$(-0.5 + 0.5\lambda)\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	1	*	$-3(1 - \mu_s) + (3(1 - \mu_s) + 1)\lambda$
V ₂	\mathbf{M}^c	0_L	$(-0.5 + 0.5\lambda)\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	0	*	$-3(1 - \mu_s) + (3(1 - \mu_s) + 1)\lambda$
V ₃	\mathbf{M}^c	0_L	$(-0.5 + 0.5\lambda)\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	1	*	$-3(1 - \mu_s) + (3(1 - \mu_s) + 1)\lambda$
V ₄	\mathbf{M}^c	0_L	$(2.5 - 0.5\lambda - \lambda^{-1})\mathbf{I}$	$\mathbf{v}_1 \rightarrow 0_L$
	*	$(2 - \lambda^{-1})$	*	$-3(1 - \mu_s) + (3(1 - \mu_s) + 1)\lambda$

Tableau III.4: Matrices $\mathbf{F}_{22}(n)$ et $\mathbf{F}_{33}(n)$ en régime asymptotique pour le choix

\mathbf{C}_4 ($\mu^\gamma = 0$, $\mu^\beta = 1.5$, $\mu^b = 0.5$); $0 \leq \mu_s \leq 1$; La quantité (*) n'influence pas sur l'étude.

Pour un facteur d'oubli $\lambda < 1$; nous remarquons que les matrices \mathbf{F}_{33} de tous les cas (choix, versions) citées dans les tableaux (III.1 à III.4) ont leurs valeurs propres strictement inférieures à 1. Par contre, les valeurs propres des matrices \mathbf{F}_{22} ne le sont pas pour tous les cas. Le gain scalaire asymptotique c_γ déduit de l'expression (III.20a) est égal à 1 pour les versions V_1 et V_3 des choix $\mu^\gamma = 0$ ou $\mu_s = 0$. Ce système se réduit alors approximativement à un système de type marche aléatoire. Par conséquent, on évitera ce cas.

L'analyse utilisant le comportement moyen du système (III.6) ne permet pas de trouver la véritable condition de stabilité, car les erreurs numériques peuvent être de moyenne nulle mais de variance non bornée.

Calculons la variance de $\Delta \mathbf{b}_n$ notée $\sigma_{\Delta b, n}^2$, pour cela on utilise l'approche statistique.

$$\mathbf{E}\{\Delta \mathbf{b}_n \Delta \mathbf{b}_n^T\} = \sigma_{\Delta b, n}^2 \mathbf{I} \quad (\text{III.21})$$

Nous pouvons écrire :

$$\begin{aligned} \mathbf{E}\{\Delta \mathbf{b}_n \Delta \mathbf{b}_n^T\} &= \mathbf{E}\{(\theta_n^{-1})^2\} \mathbf{E}\{\Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T\} - \mathbf{E}\{\theta_n^{-1} c_b(n)\} \mathbf{E}\{\mathbf{R}_{N,t}^{-1} \mathbf{x}_n \mathbf{x}_n^T\} \mathbf{E}\{\Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T\} \\ &- \mathbf{E}\{\theta_n^{-1} c_b(n)\} \mathbf{E}\{\Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T\} \mathbf{E}\{\mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T\}^T + \mathbf{E}\{c_b^2(n)\} \mathbf{E}\{\mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T (\mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T)^T\} \\ &+ \mathbf{E}\{\mathbf{p}_b(n) \mathbf{p}_b^T(n)\} \end{aligned} \quad (\text{III.22})$$

Nous supposons alors que les éléments du vecteur perturbateur $\mathbf{p}_b(n)$ sont des séquences de bruit blanc centré et de variance connue $\sigma_{p_b}^2$. En utilisant les approximations citées plus haut, nous trouvons :

$$\begin{aligned} \mathbf{E}\{\Delta \mathbf{b}_n \Delta \mathbf{b}_n^T\} &= \lambda^{-2} \mathbf{E}\{\Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T\} - 2\lambda^{-1} (1-\lambda) c_b \mathbf{E}\{\Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T\} \\ &+ c_b^2 (1-\lambda) c_b^2 \mathbf{R}_{xx}^{-1} \mathbf{E}\{\mathbf{x}_n \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T \mathbf{x}_n \mathbf{x}_n^T\} (1-\lambda) (\mathbf{R}_{xx}^{-1})^T + \mathbf{E}\{\mathbf{p}_b(n) \mathbf{p}_b^T(n)\} \end{aligned} \quad (\text{III.23})$$

avec

$$c_b = \begin{cases} (\mu^b - \mu^\gamma) + (1 + \mu^\gamma) \lambda^{-1} ; & V_{1,3} \\ \mu^b + \lambda^{-1} ; & V_2 \\ (\mu^b - 2\mu^\gamma - 1) + 2(1 + \mu^\gamma) \lambda^{-1} ; & V_4 \end{cases} \quad (\text{III.24})$$

qui peut se mettre sous la forme générale $(\mu_0 + \mu_1 \lambda^{-1})$ où μ_0 et μ_1 peuvent être facilement déduites de (III.24).

En plus, nous supposons que les composantes du vecteur $\Delta \mathbf{b}_n$ sont indépendantes entre elles et indépendantes des différentes variables théoriques de l'algorithme. Et en supposant en plus que

le signal d'entrée est une séquence de variables aléatoires gaussiennes décorrélée de variance σ_x^2 , ce qui permet d'écrire $E\{\mathbf{x}_n \mathbf{x}_n^T\} = \mathbf{R}_{xx} = \sigma_x^2 \mathbf{I}$. L'expression (III.23) devient :

$$\sigma_{\Delta b, n}^2 \mathbf{I} = \lambda^{-2} \sigma_{\Delta b, n-1}^2 \mathbf{I} - 2\lambda^{-1} (1-\lambda) c_b \sigma_{\Delta b, n-1}^2 \mathbf{I} + c_b^2 (1-\lambda)^2 \sigma_x^{-4} \mathbf{H}(n) + \sigma_{p_b}^2 \mathbf{I} \quad (\text{III.25})$$

où

$$\mathbf{H}(n) = E\{\mathbf{x}_n \mathbf{x}_n^T \Delta \mathbf{b}_{n-1} \Delta \mathbf{b}_{n-1}^T \mathbf{x}_n \mathbf{x}_n^T\} \quad (\text{III.26})$$

est une matrice carré d'ordre L , dont les éléments s'écrivent :

$$h_{ij} = \sum_{k=1}^L \sum_{m=1}^L E\{\Delta b_{k, n-1} \Delta b_{m, n-1} x_{n-k+1} x_{n-m+1} x_{n-i+1} x_{n-j+1}\} \quad (\text{III.27})$$

En développant cette dernière relation, nous trouvons :

$$h_{ij} = \sum_{k=1}^L \left[\sigma_{\Delta b, n-1}^2 E\{x_{n-k+1}^2\} E\{x_{n-i+1} x_{n-j+1}\} + \sigma_{\Delta b, n-1}^2 E\{x_{n-k+1} x_{n-i+1}\} E\{x_{n-i+1} x_{n-j+1}\} \right. \\ \left. + \sigma_{\Delta b, n-1}^2 E\{x_{n-k+1} x_{n-j+1}\} E\{x_{n-i+1} x_{n-j+1}\} \right] \quad (\text{III.28})$$

De la relation (III.28), nous pouvons dire que si :

$$i \neq j \Rightarrow h_{ij} = 0$$

$$i = j \Rightarrow h_{ii} = \sigma_{\Delta b, n-1}^2 \sigma_x^2 \sum_{k=1}^L \left[E\{x_{n-k+1}^2\} + 2E\{x_{n-k+1} x_{n-i+1}\} \right] \quad (\text{III.29})$$

Enfin, nous pouvons écrire :

$$h_{ij} = \begin{cases} \sigma_{\Delta b, n-1}^2 \sigma_x^4 (L+2) & i = j \\ 0 & i \neq j \end{cases} \quad (\text{III.30})$$

Finalement, nous obtenons l'expression suivante :

$$\sigma_{\Delta b, n}^2 = G \sigma_{\Delta b, n-1}^2 + \sigma_{p_b}^2 \quad (\text{III.31})$$

où

$$G = \left[(1 - 2\mu_1 + 2\mu_1^2 + \mu_1^2 L) + 2(\mu_0 - \mu_1)(\mu_1(L+2) - 1)\lambda \right. \\ \left. + (2\mu_0 + (\mu_0^2 + \mu_1^2 - 4\mu_0\mu_1)(L+2))\lambda^2 + 2\mu_0(\mu_1 - \mu_0)(L+2)\lambda^3 + \mu_0^2(L+2)\lambda^4 \right] \lambda^{-2} \quad (\text{III.32})$$

L'étude de la stabilité de (III.31) (pôle à l'intérieur du cercle unité) est la bornitude de $\sigma_{\Delta b, n}^2$.

En supposant que $\sigma_{p_b}^2$ est bornée, la condition de stabilité de la relation (III.31) est donnée par la solution de l'inégalité suivante :

$$|G| < 1 \quad (\text{III.33})$$

Pour simplifier les calculs, prenons des choix simples et convenables des paramètres de contrôle qui satisfont les conditions citées plus haut :

$$C_1: (\mu^\gamma = \mu^\beta = \mu^b = 1); 0 \leq \mu_s \leq 1 \text{ pour } V_{1,2,3} \quad (\text{III.34a})$$

$$C_2: (\mu^\gamma = 0, \mu^\beta = \mu^b = 1); 0 \leq \mu_s \leq 1 \text{ pour } V_{1,2,3,4} \quad (\text{III.34b})$$

L'expression (III.32) est fonction seulement que de λ et L . En appliquant la condition (III.33), il vient :

$$(C_1, V_{1,3}), (C_2, V_4), \Rightarrow \lambda > \frac{4L+5}{4L+7} = 1 - \frac{1}{2L+3.5} \quad (\text{III.35a})$$

$$(C_1, V_2), (C_2, V_{1,2,3}) \Rightarrow \lambda > \sqrt{\frac{L+1}{L+2}} = \left(1 - \frac{1}{L+2}\right)^{1/2} \quad (\text{III.35b})$$

Ces conditions peuvent être écrites sous une forme plus simple :

$$\lambda > 1 - \frac{1}{pL} \quad (\text{III.36})$$

où le paramètre p est un réel strictement supérieur à 2 pour assurer la stabilité numérique.

III.3 Analyse de convergence

L'analyse est basée sur l'hypothèse d'indépendance entre le vecteur du signal d'entrée \mathbf{x}_n et le vecteur des coefficients du filtre adaptatif \mathbf{w}_n . On supposera que le système à identifier peut être modélisé par un filtre linéaire \mathbf{w}_{opt} . La réponse à ce système est le résultat de la convolution entre ce filtre et le signal d'excitation x_n c'est-à-dire $y_n = \mathbf{w}_{opt}^T \mathbf{x}_n$, à ce signal vient s'ajouter un bruit v_n ; ce qui donne le signal désiré d_n . Le signal estimé est donné par la convolution $\hat{y}_n = \mathbf{w}_{n-1}^T \mathbf{x}_n$.

Le signal d'erreur en sortie du filtre adaptatif est (Fig. III.1) :

$$\bar{\varepsilon}_n = d_n - \hat{y}_n = v_n + \mathbf{x}_n^T (\mathbf{w}_{opt} - \mathbf{w}_n) \quad (\text{III.37})$$

La mise à jour des paramètres du filtre adaptatif est :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{g}_n \bar{\varepsilon}_n \quad (\text{III.38})$$

où $\mathbf{g}_n = \mathbf{R}_n^{-1} \mathbf{x}_n$ est le gain d'adaptation.

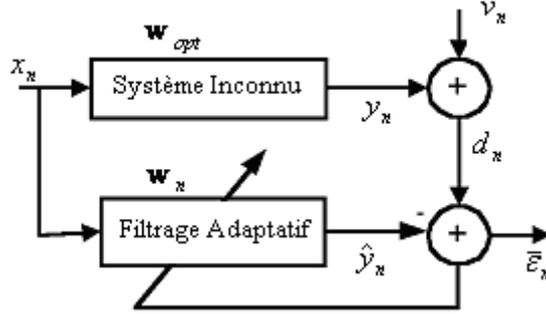


Fig.III.1: Système de filtrage FIR adaptatif

III.3.1 Convergence en moyenne

Nous définissons le vecteur d'erreur des coefficients du filtre adaptatif à l'instant n par [54]:

$$\Delta \mathbf{w}_n = \mathbf{w}_{opt} - \mathbf{w}_n \quad (III.39)$$

L'erreur de sortie $\bar{\varepsilon}_n$ s'écrit alors:

$$\bar{\varepsilon}_n = v_n + \mathbf{x}_n^T \Delta \mathbf{w}_{n-1} \quad (III.40)$$

La récurrence de (III.38) sur le vecteur d'erreur des coefficients $\Delta \mathbf{w}_n$ donne :

$$\Delta \mathbf{w}_n = [\mathbf{I}_L - \mathbf{g}_n \mathbf{x}_n^T] \Delta \mathbf{w}_{n-1} - \mathbf{g}_n v_n \quad (III.41)$$

En prenant l'espérance des deux termes de la relation (III.41), on peut écrire :

$$E\{\Delta \mathbf{w}_n\} = E\{[\mathbf{I}_L - \mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T] \Delta \mathbf{w}_{n-1}\} - E\{\mathbf{R}_n^{-1} \mathbf{x}_n v_n\} \quad (III.42)$$

Le bruit v_n est de moyenne nulle et a pour variance σ_v^2 . En exploitant l'hypothèse d'indépendance, on trouve :

$$E\{\Delta \mathbf{w}_n\} = \lambda E\{\Delta \mathbf{w}_{n-1}\} \quad (III.43)$$

En régime permanent, lorsque le filtre a convergé, la solution stable de (III.43) est : si $\lambda < 1 \Rightarrow$

$$E\{\Delta \mathbf{w}(\infty)\} = \mathbf{0}_L.$$

D'où le vecteur moyen des coefficients du filtre adaptatif est :

$$E\{\mathbf{w}(\infty)\} = \mathbf{w}_{opt} \quad (III.44)$$

L'algorithme FRLS converge en moyenne vers la solution optimale.

III.3.2 Convergence en quadratique moyenne

L'erreur quadratique moyenne (MSE) peut s'écrire :

$$\begin{aligned} MSE(n) &= E\{\bar{\varepsilon}_n^2\} = E\{(v_n + \mathbf{x}_n^T \Delta \mathbf{w}_{n-1})^2\} = \sigma_v^2 + \text{tr}[\mathbf{R}_{xx} E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\}] \\ &= \sigma_v^2 + \sigma_x^2 E\{\|\Delta \mathbf{w}_{n-1}\|^2\} \end{aligned} \quad (III.45)$$

où $\text{tr}[\cdot]$ et $\|\cdot\|$ représentent respectivement la trace d'une matrice et la norme euclidienne d'un vecteur. Nous définissons la norme du vecteur d'erreur des coefficients normalisée par [54]:

$$NM(n) = 10 \log_{10} \left(\frac{E\{\|\Delta \mathbf{w}_n\|^2\}}{\|\mathbf{w}_{opt}\|^2} \right) \quad (\text{III.46})$$

qui mesure la disparité entre la réponse impulsionnelle réelle et le filtre modélisé. Pour cela, nous devons déterminer les prochaines expressions :

$$E\{\Delta \mathbf{w}_n \Delta \mathbf{w}_n^T\} = E\left\{ \left([\mathbf{I} - \mathbf{g}_n \mathbf{x}_n^T] \Delta \mathbf{w}_{n-1} - \mathbf{g}_n v_n \right) \left([\mathbf{I} - \mathbf{g}_n \mathbf{x}_n^T] \Delta \mathbf{w}_{n-1} - \mathbf{g}_n v_n \right)^T \right\} \quad (\text{III.47})$$

En développant cette dernière, on obtient :

$$E\{\Delta \mathbf{w}_n \Delta \mathbf{w}_n^T\} = E\left\{ [\mathbf{I} - \mathbf{g}_n \mathbf{x}_n^T] \Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T [\mathbf{I} - \mathbf{g}_n \mathbf{x}_n^T]^T \right\} + E\{\mathbf{g}_n \mathbf{g}_n^T v_n^2\} \quad (\text{III.48})$$

En utilisant les hypothèses d'indépendances, on obtient :

$$\begin{aligned} E\{\Delta \mathbf{w}_n \Delta \mathbf{w}_n^T\} &= E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\} - E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\} E\{\mathbf{g}_n \mathbf{x}_n^T\}^T - E\{\mathbf{g}_n \mathbf{x}_n^T\} E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\} \\ &\quad + E\{\mathbf{g}_n \mathbf{x}_n^T \Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T \mathbf{g}_n \mathbf{x}_n^T\} + E\{\mathbf{g}_n \mathbf{g}_n^T\} E\{v_n^2\} \end{aligned} \quad (\text{III.49})$$

où

$$E\{\mathbf{g}_n \mathbf{g}_n^T\} = E\{\mathbf{R}_n^{-1} \mathbf{x}_n \mathbf{x}_n^T \mathbf{R}_n^{-1}\} = (1 - \lambda)^2 \mathbf{R}_{xx}^{-1} \quad (\text{III.50})$$

Ainsi l'expression (III.49) devient :

$$E\{\Delta \mathbf{w}_n \Delta \mathbf{w}_n^T\} = \lambda^2 E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\} + (1 - \lambda)^2 \frac{\sigma_v^2}{\sigma_x^2} \mathbf{I} \quad (\text{III.51})$$

En prenant la trace des deux termes de la relation (III.51), on peut écrire :

$$\text{tr}[E\{\Delta \mathbf{w}_n \Delta \mathbf{w}_n^T\}] = \lambda^2 \text{tr}[E\{\Delta \mathbf{w}_{n-1} \Delta \mathbf{w}_{n-1}^T\}] + (1 - \lambda)^2 \frac{\sigma_v^2}{\sigma_x^2} L \quad (\text{III.52})$$

Finalement, la moyenne de la norme au carré du vecteur d'erreur des coefficients du filtre adaptatif est :

$$E\{\|\Delta \mathbf{w}_n\|^2\} = \lambda^2 E\{\|\Delta \mathbf{w}_{n-1}\|^2\} + (1 - \lambda)^2 \frac{\sigma_v^2}{\sigma_x^2} L \quad (\text{III.53})$$

La stabilité de la récurrence (III.53) est garantie si $\lambda < 1$.

III.4 Résultats de simulation

III.4.1 Comportement des algorithmes FRLS avec des signaux stationnaire

Pour confirmer la validité de notre analyse et démontrer l'exécution numérique améliorée, des simulations sont effectuées. Le signal d'entrée utilisé est un bruit blanc gaussien, de moyenne nulle et de variance unité.

III.4.1.1 Partie prédiction

Nous nous intéressons ici qu'au bon fonctionnement de la partie prédiction aller/retour des algorithmes modifiés car les divergences concernent uniquement la partie prédiction ; la partie filtrage est robuste par rapport à l'implantation numérique, elle nécessite seulement un facteur d'oubli inférieur à 1 [55] et le bon fonctionnement du gain d'adaptation (gain de Kalman) fourni par la partie prédiction. Elle est stable si cette dernière l'est. Pour cela, on évalue le bon comportement de la variable indicateur de divergence ($\xi_n \rightarrow 0$) et la variable de vraisemblance ($0 < \gamma_n \leq 1$). Nous évaluons les erreurs numériques $\sigma_{\xi}^2(n)$, donnée par (II.39). Cette variable est utilisée dans nos simulations dans un but de comparaison pour un meilleur choix entre les versions. Nous avons simulé les algorithmes pour vérifier leurs exactitudes. Pour un choix sélectionné (III.34) et un facteur d'oubli λ (III.36), pris dans de bonne condition de stabilité numérique ($p > 2$), nous comparons ici les nouvelles versions ($V_{1, 2, 3, 4}$) entre elles (Fig. III.3a à III.3d) et avec les versions V_B [51] et V_S [50] (Fig. III.4). Le filtre est d'ordre $L=32$, le facteur d'oubli est $\lambda = 0.9896$ ($p=3$). Des simulations ont été faites pour plus de 10^7 échantillons.

Nous notons que les simulations (Fig. III.3b) donnent les mêmes résultats pour les cas C_1V_2 et C_2V_2 , et de même pour les cas C_3V_2 et C_4V_2 , parce que la version V_2 ne dépend pas de μ^γ .

Les cas C_2V_1 et C_4V_1 (Fig. III.3a) ainsi que C_2V_3 et C_4V_3 (Fig. III.3c) sont les moins bons des autres cas, ceci est dû à $c_\gamma = 1$ ($\mu^\gamma = 0$) qui ne satisfait pas strictement à la condition $c_\gamma < 1$.

Ces observations confirment la validité de notre analyse présentée dans la section III.2.2.

La dynamique de la propagation d'erreur est la même dans tous les algorithmes, mais le mécanisme de produire des erreurs approximatives est différent. Une variance inférieure de ces erreurs approximatives produites mène à de plus petites erreurs approximatives cumulées et, par conséquent, à une meilleure exactitude. D'après les résultats de simulation effectués, dans les conditions de stabilité, en général il est très difficile, de dire quel est le meilleur de ces algorithmes, parce qu'il y a trop de variables mises en jeu (Fig. III.4). Le fait nouveau ramené par ces nouvelles versions d'algorithmes est qu'elles évitent la marche aléatoire dans le calcul de la

variable de vraisemblance. Mais en comparant avec les erreurs numériques des versions V_B et V_S , on remarque que la version C_3V_1 est plus précise (Fig. III.5).

Nous définissons par la suite, le rapport des carrés des normes des gains en dB par [56]:

$$NGE(n) = 10 \log_{10} \left(\frac{E\{\|\Delta \mathbf{g}_n\|^2\}}{E\{\|\mathbf{g}_n\|^2\}} \right) \quad (\text{III.56})$$

où $\Delta \mathbf{g}_n = (\mathbf{R}_n^{-1} \mathbf{x}_n - \gamma_n \tilde{\mathbf{k}}_n)$ est le vecteur erreur du gain. La quantité $NGE(n)$ permet de mesurer la disparité entre les gains $\mathbf{R}_n^{-1} \mathbf{x}_n$ et $\gamma_n \tilde{\mathbf{k}}_n$ calculés par les algorithmes RLS et NS-FRLS respectivement. Pour un choix convenable des paramètres de contrôle, nous avons vérifié la validité des conditions de stabilité numérique sur λ donné par l'équation (III.36). En outre, si cette condition n'est pas satisfaite alors l'algorithme proposé diverge.

Les figures (III.6) et (III.7), montrent l'évolution des variables γ_n , ξ_n et $NGE(n)$ correspondant au cas numériquement instable ($p < 2$) et numériquement stable ($p > 2$) respectivement.

Pendant les premières itérations (Fig.III.6a), la variable de vraisemblance γ_n est presque constante (valeur asymptotique), alors elle commence à osciller, pour finir avec une divergence. L'indicateur de divergence ξ_n augmente indéfiniment jusqu'à la divergence totale de l'algorithme. Pour le rapport des carrés des normes des gains $NGE(n)$, nous notons qu'il diverge bien avant les autres variables; cette courbe croît exponentiellement et arrive au niveau de la variance du signal d'entrée, à ce moment l'algorithme a déjà divergé.

Le schéma (Fig. III.7) illustre la stabilité de cette version, où l'indicateur de divergence ξ_n reste très faible, et la variable de vraisemblance γ_n flotte autour de sa valeur optimale. Nous notons que la variable $NGE(n)$ converge et reste stable. Nous pouvons dire donc, que l'algorithme FRLS modifié est bien stabilisé numériquement.

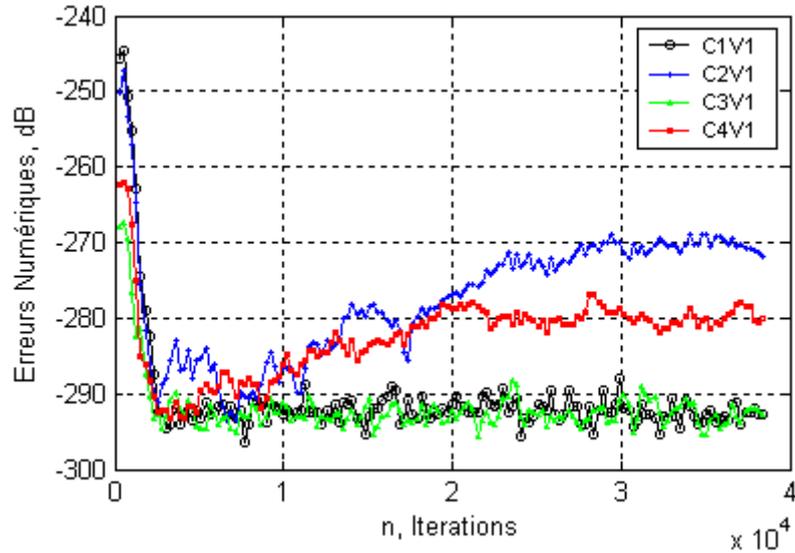


Fig. III.3a: Evolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version V_1 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5$, $L=32$, $\lambda=0.9896$ ($p=3$)

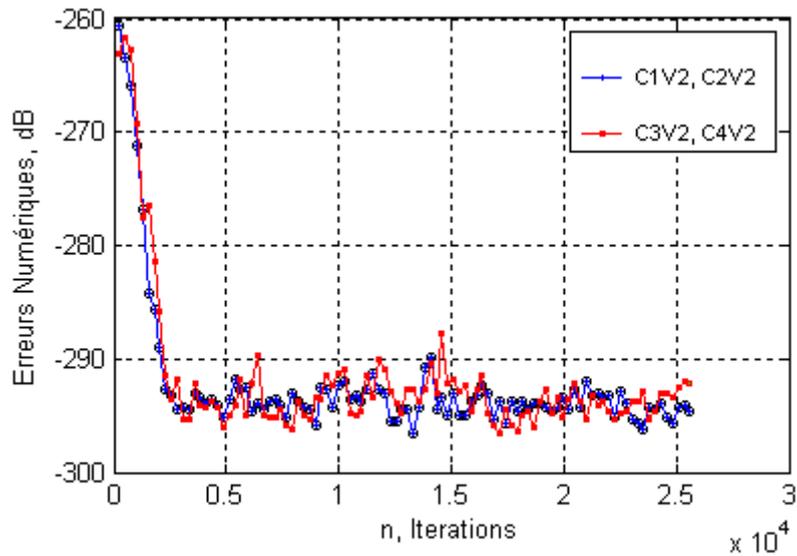


Fig. III.3b: Evolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version V_2 pour les choix $C_{1,2,3,4}$; $\mu_s=0.5$, $L=32$, $\lambda=0.9896$ ($p=3$)

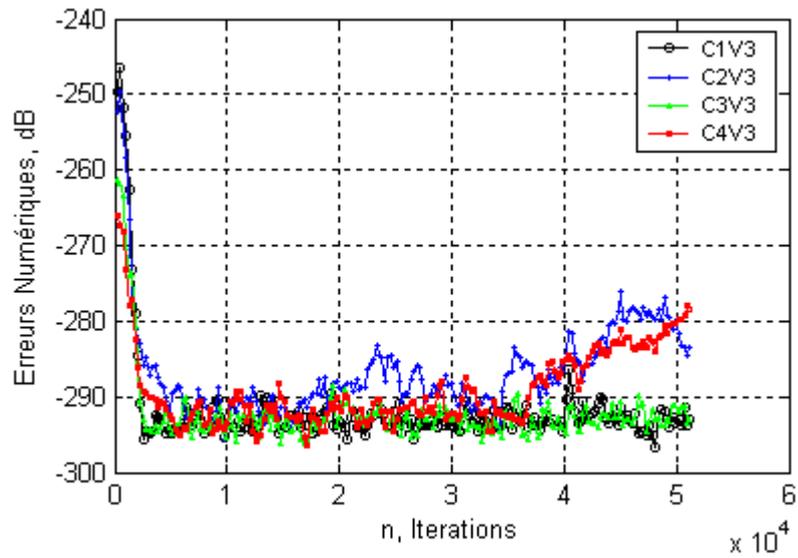


Fig. III.3c: Evolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version V_3 pour le choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)

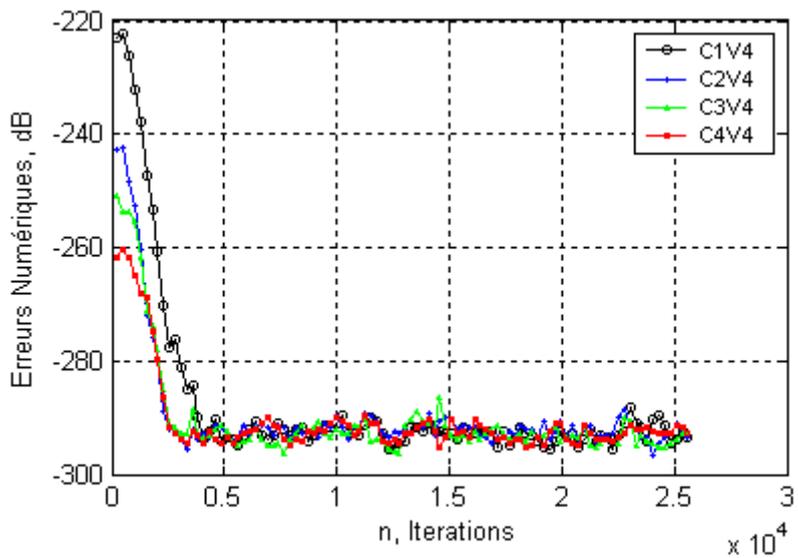


Fig. III.3d: Evolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version V_4 pour le choix $C_{1,2,3,4}$; $\mu_s=0.5, L=32, \lambda=0.9896$ ($p=3$)

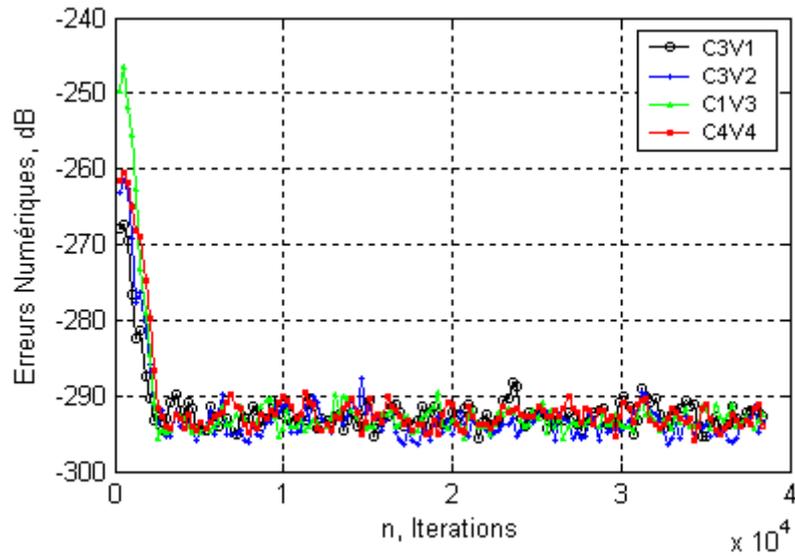


Fig. III.4: Comparaison de l'évolution des erreurs numériques $\sigma_{\xi}^2(n)$ des meilleures versions pour $\mu_s=0.5, L=32, \lambda=0.9896 (p=3)$

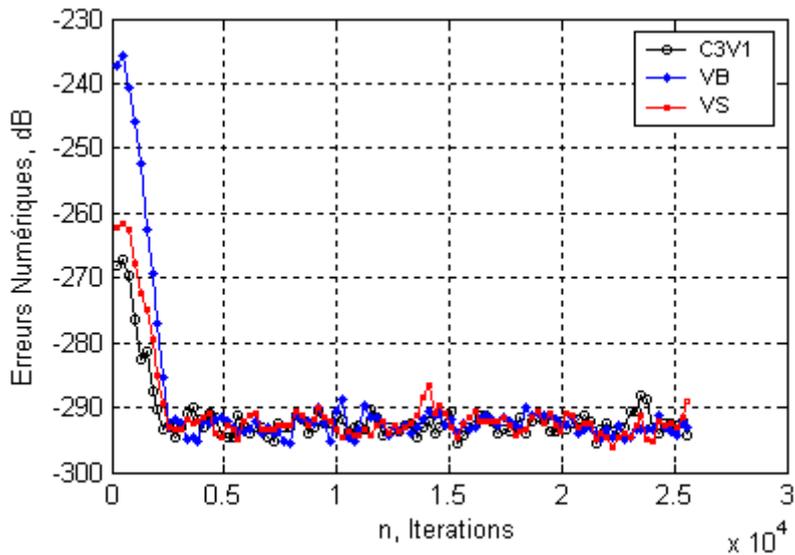


Fig. III.5: Comparaison de l'évolution des erreurs numériques $\sigma_{\xi}^2(n)$ de la version C_3V_1 avec les versions V_B et V_S pour $L=32, \lambda=0.9896 (p=3)$

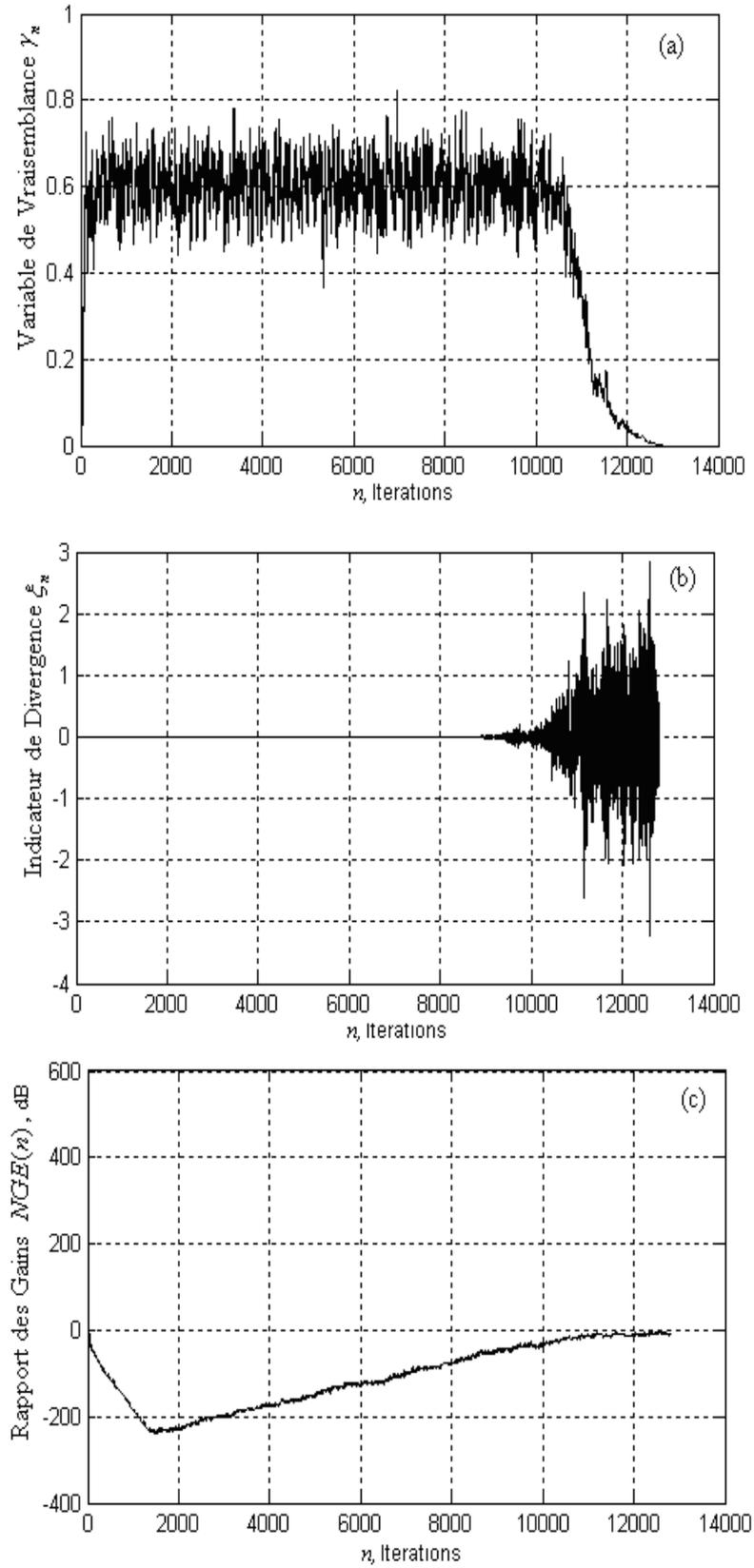


Fig.III.6: Evolution des variables γ_n , ξ_n et $NGE(n)$
 du cas $(C_3V_1, \mu_s=0.5)$ pour $L=32, p=1.95$

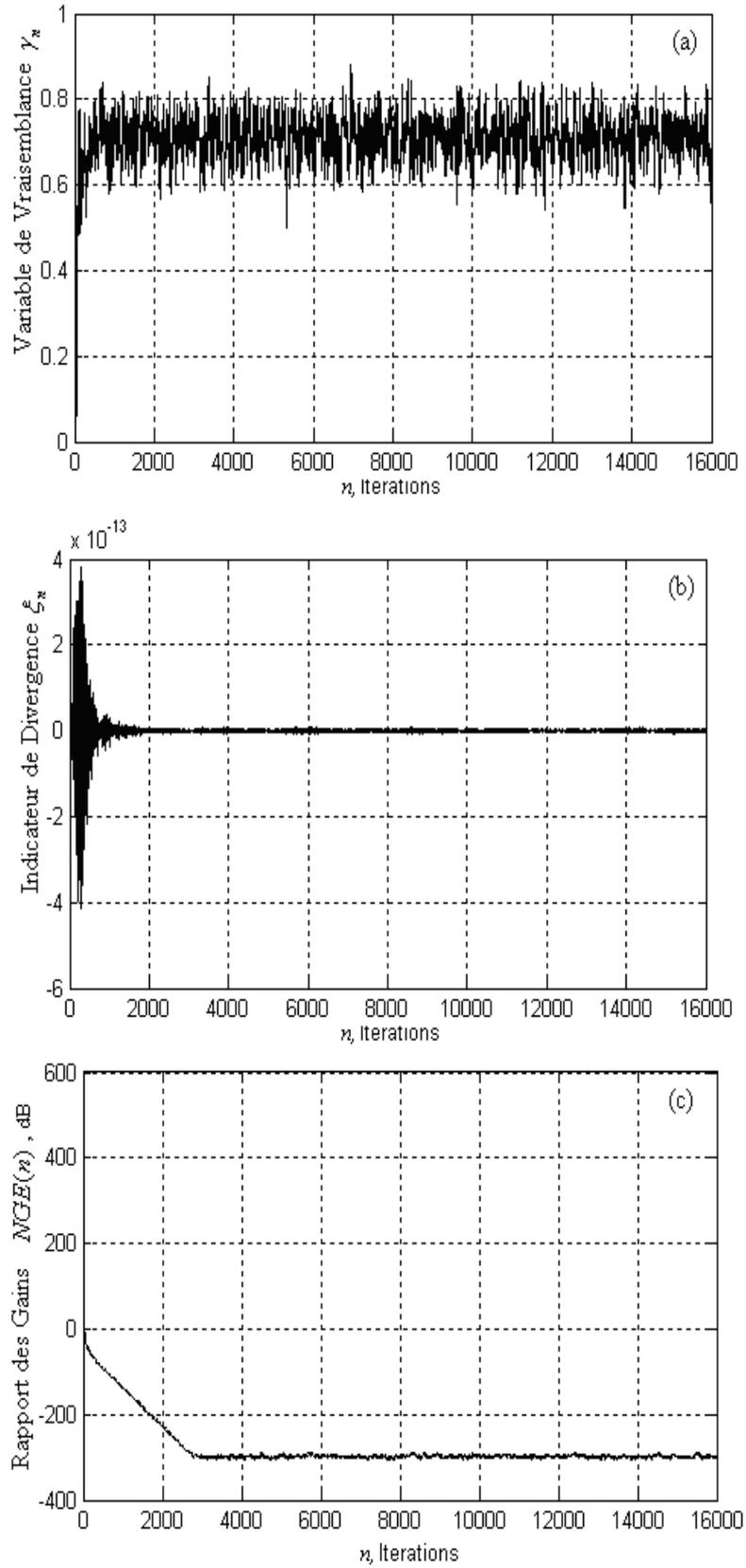


Fig.III.7: Evolution des variables γ_n , ξ_n et $NGE(n)$
 du cas $(C_3V_1, \mu_s=0.5)$ pour $L=32, p=3$

III.4.1.2 Partie filtrage

Nous essayons d'estimer une réponse impulsionnelle \mathbf{w}_{opt} de longueur $L=32$ ($L=256$), la même longueur est utilisée pour le filtre adaptatif \mathbf{w}_n . La performance d'évaluation est mesurée par l'erreur quadratique moyenne $MSE(n)$ donnée par (III.45) et la norme du vecteur d'erreur des coefficients normalisée $NM(n)$ donnée par (III.46). Le signal de référence d_n est obtenu par convolution de \mathbf{w}_{opt} avec le signal d'entrée x_n et en additionnant un bruit blanc gaussien de rapport signal à bruit SNR de 50dB. On fait tourner l'algorithme NS-FRLS avec un facteur d'oubli $\lambda = (1 - 1/3L)$.

La figure III.8, présente le $MSE(n)$ déterminé par simulation et par calcul théorique (III.45). A partir de ce graphe, nous observons que les courbes de la simulation et théorique concordent très bien. La figure III.9 montre la convergence de $NM(n)$ comme obtenu à partir de l'analyse théorique (III.46) et à partir des résultats de simulation. Il peut être vu que, pour un filtre adaptatif, il y a une bonne concordance entre le comportement réel de l'algorithme et celui prédit par l'expression théorique.

Nous avons testé cet algorithme modifié NS-FRLS de type FTF avec succès sur de très longues séquences de signaux stationnaires et avec différents ordres de filtre L .

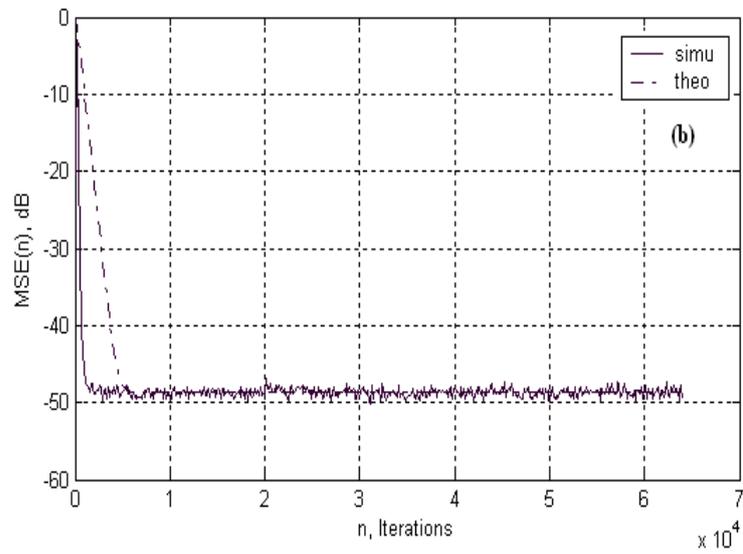
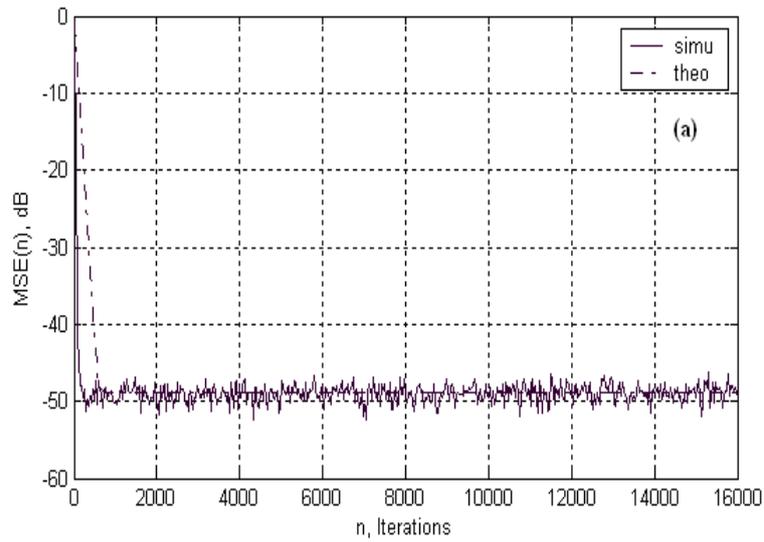


Fig.III.8: Comparaison des courbes théorique et de simulation de l'erreur quadratique moyenne $MSE(n)$ de $(C_3V_1, \mu_s=0.5)$ pour $p=3$.

(a): $L=32, \lambda=0.9896, E_0=0.5$;

(b): $L=256, \lambda=0.9987, E_0=3$.

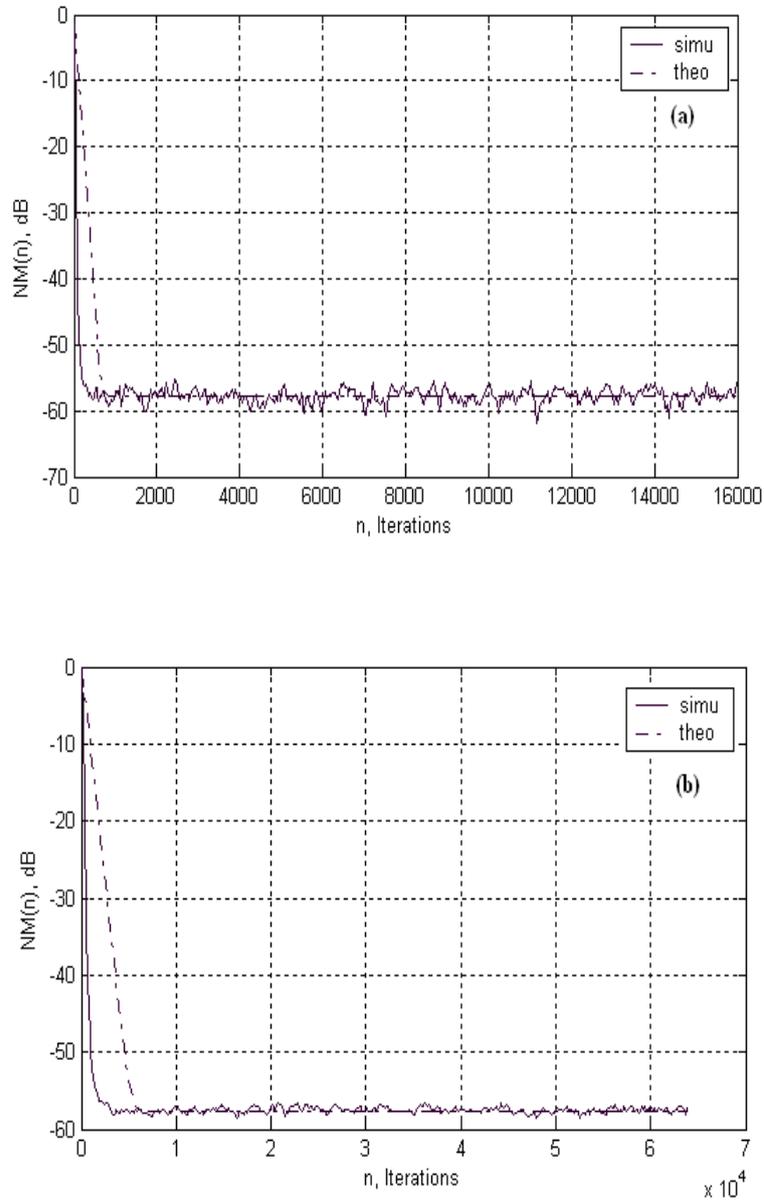


Fig.III.9: Comparaison des courbes théorique et de simulation de la norme du vecteur d'erreur des coefficients normalisée $NM(n)$ de $(C_3V_1, \mu_s=0.5)$ $p=3$.

(a): $L=32$, $\lambda=0.9896$, $E_0=0.5$;

(b): $L=256$, $\lambda=0.9987$, $E_0=3$.

III.4.2 Comportement des algorithmes FRLS avec des signaux parfaitement prédictibles

Un signal sinusoïdal pur est parfaitement modélisable par un modèle auto régressif d'ordre 2 (AR(2)). Il est représenté par :

$$x_n = a \sin(2\pi f_0 n) \quad (\text{III.57})$$

avec $f_0 = \frac{F_0}{F_e}$, sous la condition $\frac{F_0}{F_e} < \frac{1}{2}$, où F_e représente la fréquence d'échantillonnage et F_0

la fréquence de la sinusoïde.

Posons $\omega_0 = 2\pi f_0 = \frac{2\pi}{k_0}$, c'est-à-dire $F_e = k_0 F_0$ où $k_0 > 2$.

A la convergence de l'algorithme la matrice \mathbf{R}_n , définie par $\mathbf{R}_n = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T$, devient pratiquement constante, et si λ est proche de 1, elle est approximativement donnée par [57] :

$$\mathbf{R}_n^{-1} \cong (1 - \lambda) \mathbf{R}_{xx}^{-1} \quad (\text{III.58})$$

où $\mathbf{R}_{xx} = E\{\mathbf{x}_n \mathbf{x}_n^T\}$ représente la matrice d'autocorrélation du signal d'entrée.

Pour un signal sinusoïdal en entrée et un filtre d'ordre $L=2$, il vient [58]:

$$\mathbf{R}_{xx}^{-1} = \frac{2}{a^2 \sin^2(\omega_0)} \begin{bmatrix} 1 & -\cos(\omega_0) \\ -\cos(\omega_0) & 1 \end{bmatrix} \quad (\text{III.59})$$

Le vecteur gain, à la convergence, devient :

$$\mathbf{g}_n = (1 - \lambda) \mathbf{R}_{xx}^{-1} \mathbf{x}_n = \begin{bmatrix} g_{1,n} \\ g_{2,n} \end{bmatrix} \quad (\text{III.60})$$

où

$$g_{1,n} = \frac{2(1 - \lambda) \cos(\omega_0 (n-1))}{a \sin(\omega_0)} \quad (\text{III.61a})$$

$$g_{2,n} = \frac{-2(1 - \lambda) \cos(\omega_0 n)}{a \sin(\omega_0)} \quad (\text{III.61b})$$

Ainsi, les coefficients du gain oscillent avec la fréquence du signal d'entrée. Et si k_0 est très grand, $\sin(\omega_0)$ devient très petit, ces coefficients peuvent donc atteindre des valeurs très élevées.

Pour cela, Il conviendrait de prendre un k_0 pas très élevé.

En effectuant les calculs, la valeur asymptotique de la variable de vraisemblance est donnée par :

$$\gamma_n = 1 - x_n g_{1,n} - x_{n-1} g_{2,n} = (2\lambda - 1) \quad (\text{III.62})$$

Les signaux utilisés pour cette simulation sont des sinusoïdes pures ou pseudo périodiques (somme de sinusoïdes), modélisables par un ordre AR(M).

Pour un filtre d'ordre L donnée (égal, inférieur ou supérieur à l'ordre $AR(M)$) et un facteur d'oubli λ pris dans de bonnes conditions de stabilité numérique ($p > 2$) [52], nous allons tester, l'effet engendré par le choix de la fréquence d'échantillonnage par rapport à la fréquence du signal.

On choisira, pour la fréquence d'échantillonnage F_e , des valeurs multiples et non multiples de la fréquence de la sinusoïde F_0 , $F_e = k_0 F_0$, c'est-à-dire k_0 un nombre entier ou non.

Nous étudions le comportement de l'algorithme par rapport aux choix de k_0 .

Pour cela, nous évaluons le bon comportement des variables γ_n , ξ_n ($\xi_n \rightarrow 0$, $0 < \gamma_n \leq 1$) et nous observons l'évolution des modules des pôles des prédicteurs aller (ou retour).

III.4.2.1 Cas d'une sinusoïde pure

Le signal d'entrée x_n est une sinusoïde pure ($AR(2)$), pour un filtre d'ordre $L=2$, les figures III.10 et III.11 correspondent respectivement aux cas k_0 entier ($F_0 = 2kHz$, $F_e = 16kHz$) et k_0 non entier ($F_0 = 2kHz$, $F_e = 15.995kHz$).

Il est remarquable d'observer que la convergence est atteinte au bout de quelques itérations. L'estimation des pôles est très précise, ayant fourni les valeurs exactes du modèle, pour le prédicteur aller (ou retour). L'algorithme a été testé avec succès par des simulations à très long terme.

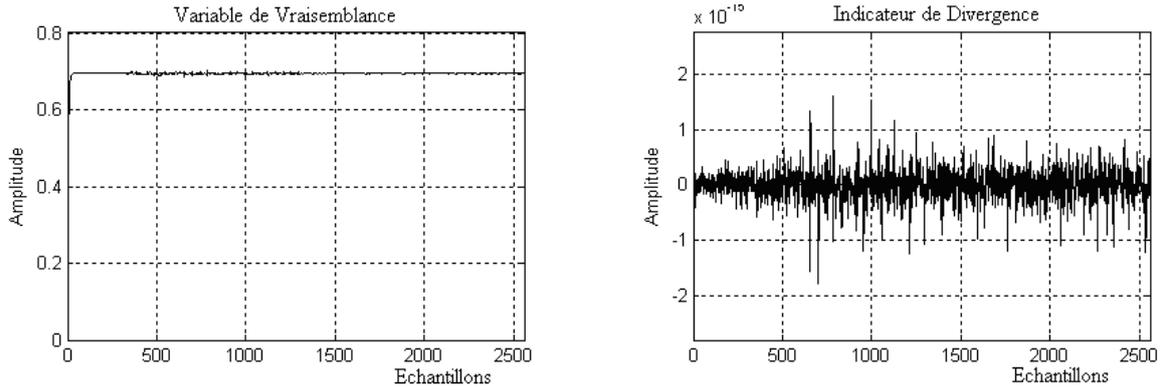


Fig.III.10a: Evolution des variables γ_n et ξ_n ($L=2$, k_0 entier)

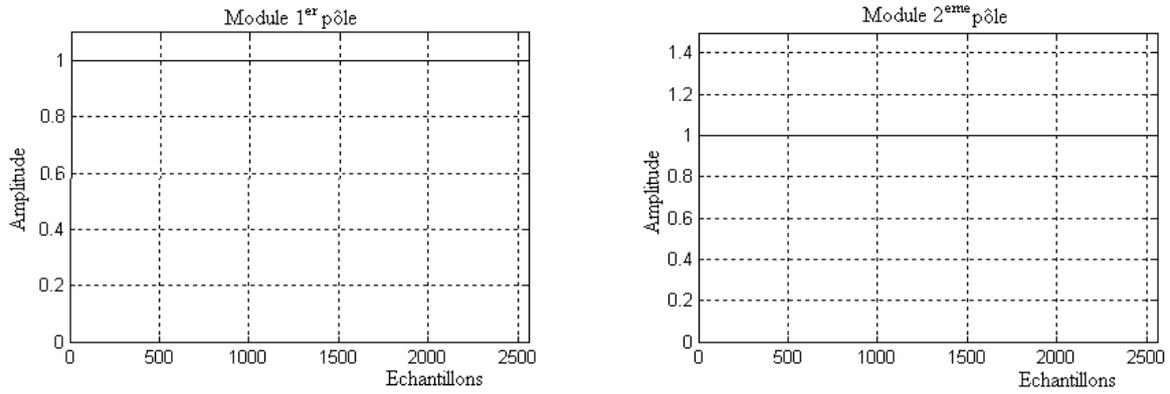


Fig.III.10b: Evolution du module des pôles du prédicteurs aller ($L=2$, k_0 entier)

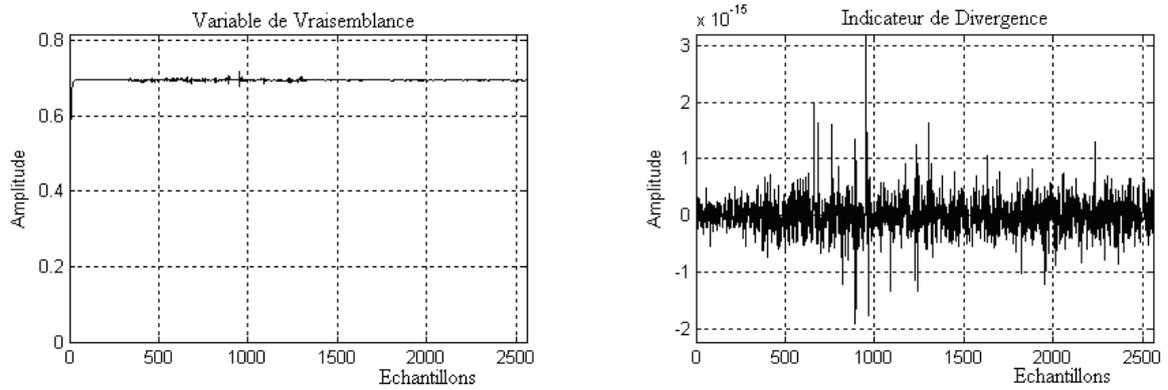


Fig.III.11a: Evolution des variables γ_n et ξ_n ($L=2$, k_0 non entier)

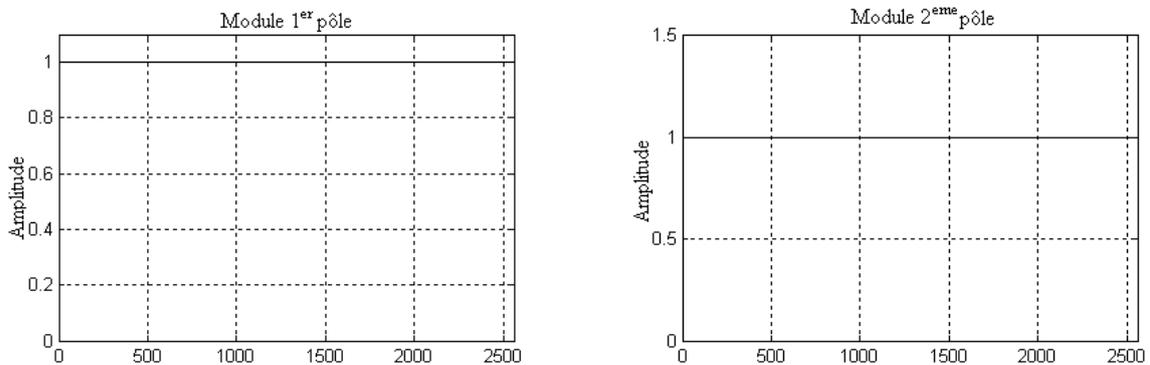


Fig.III.11b: Evolution du module des pôles du prédicteurs aller ($L=2$, k_0 non entier)

Les figures III.12a et III.12b correspondent au cas (AR(2)), un filtre d'ordre $L=3$ (surestimé) et k_0 entier ($F_0 = 2kHz, F_e = 16kHz$). Pendant les premières itérations (Fig.III.12), la variable de vraisemblance γ_n est presque constante (valeur asymptotique), puis elle commence à osciller, pour terminer à une divergence. La variable indicateur de divergence ξ_n augmente indéfiniment jusqu'à divergence totale de l'algorithme. Pour les pôles du prédicteur aller, on remarque qu'un des pôles sort du cercle unité et présente un pic de forte amplitude au moment de la détection de la divergence. Alors que les autres pôles, ne sortent pas du cercle unité, mais présentent quand même quelques oscillations avant l'instant de divergence.

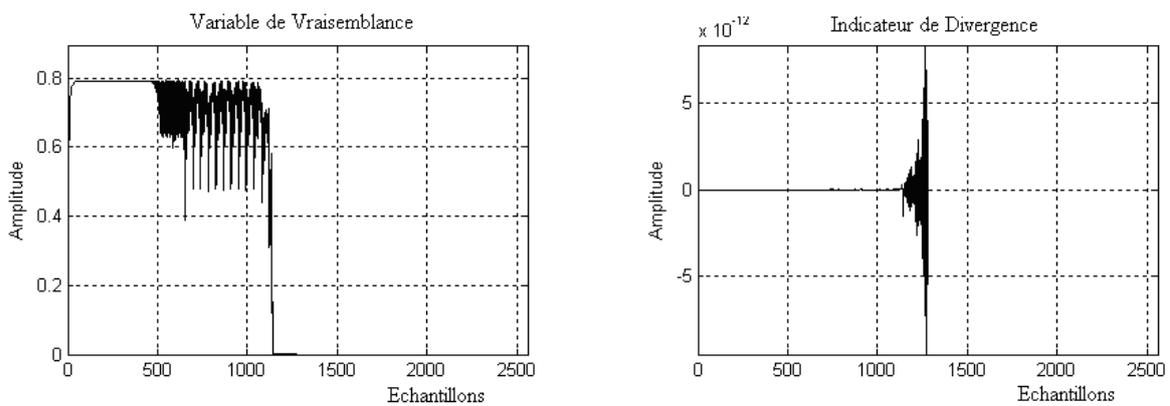


Fig.III.12a: Evolution des variables γ_n et ξ_n ($L=3, k_0$ entier)

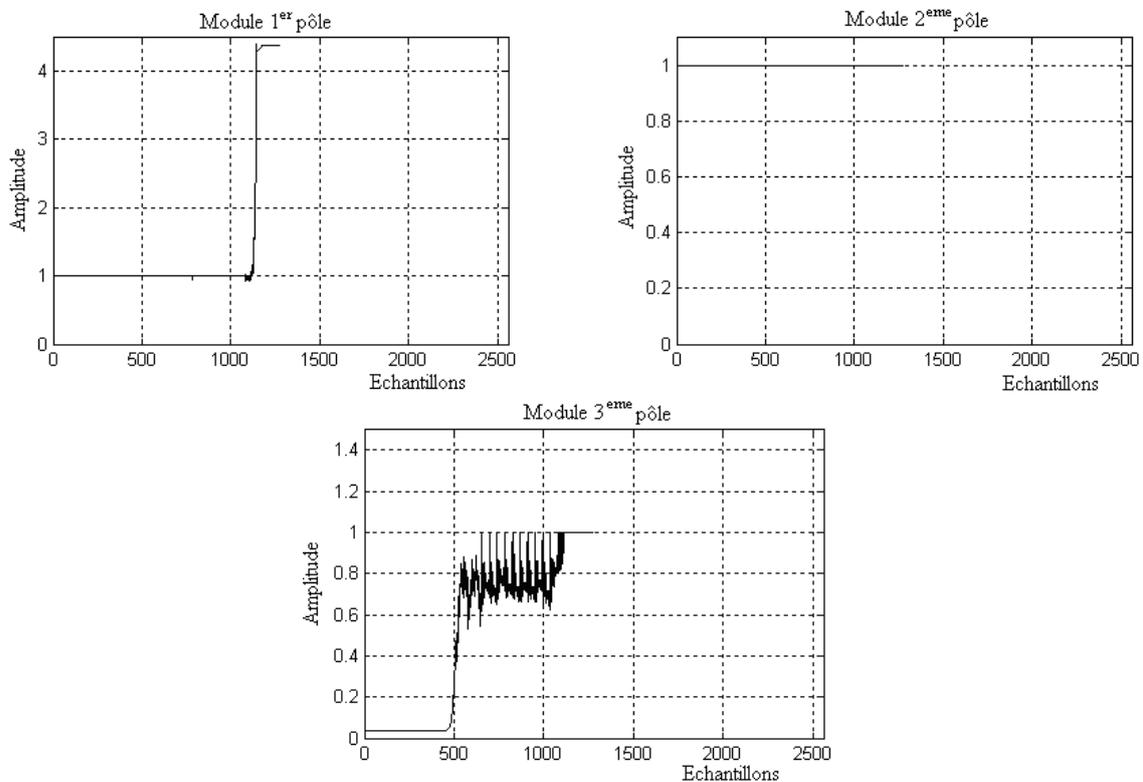


Fig.III.12b: Evolution du module des pôles du prédicteurs aller ($L=3, k_0$ entier)

Les figures III.13a et III.13b correspondent au cas (AR(2)), un filtre d'ordre $L=3$ (surestimé) et k_0 non entier ($F_0 = 2kHz, F_e = 15.995kHz$)

On remarque que malgré l'oscillation du pôle libre le long de la séquence, aucune tendance à la divergence n'a été détectée. L'algorithme reste stable à très long terme.

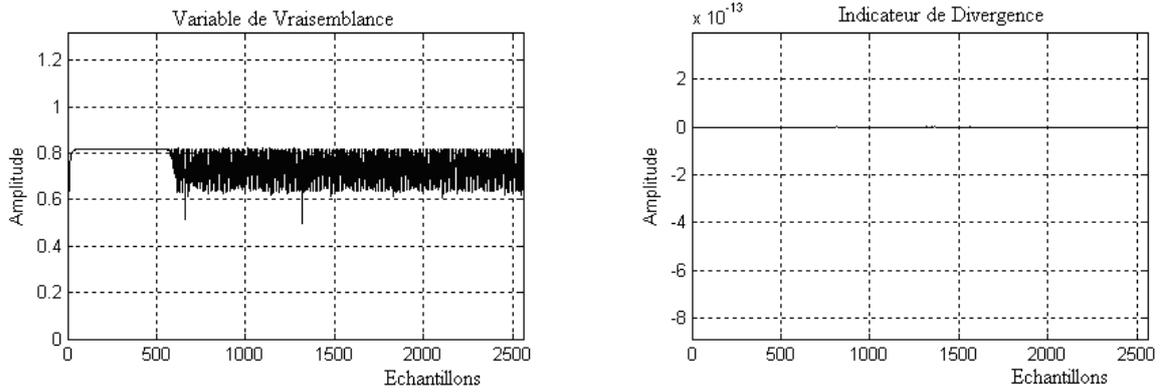


Fig.III.13a: Evolution des variables γ_n et ξ_n ($L=3, k_0$ non entier)

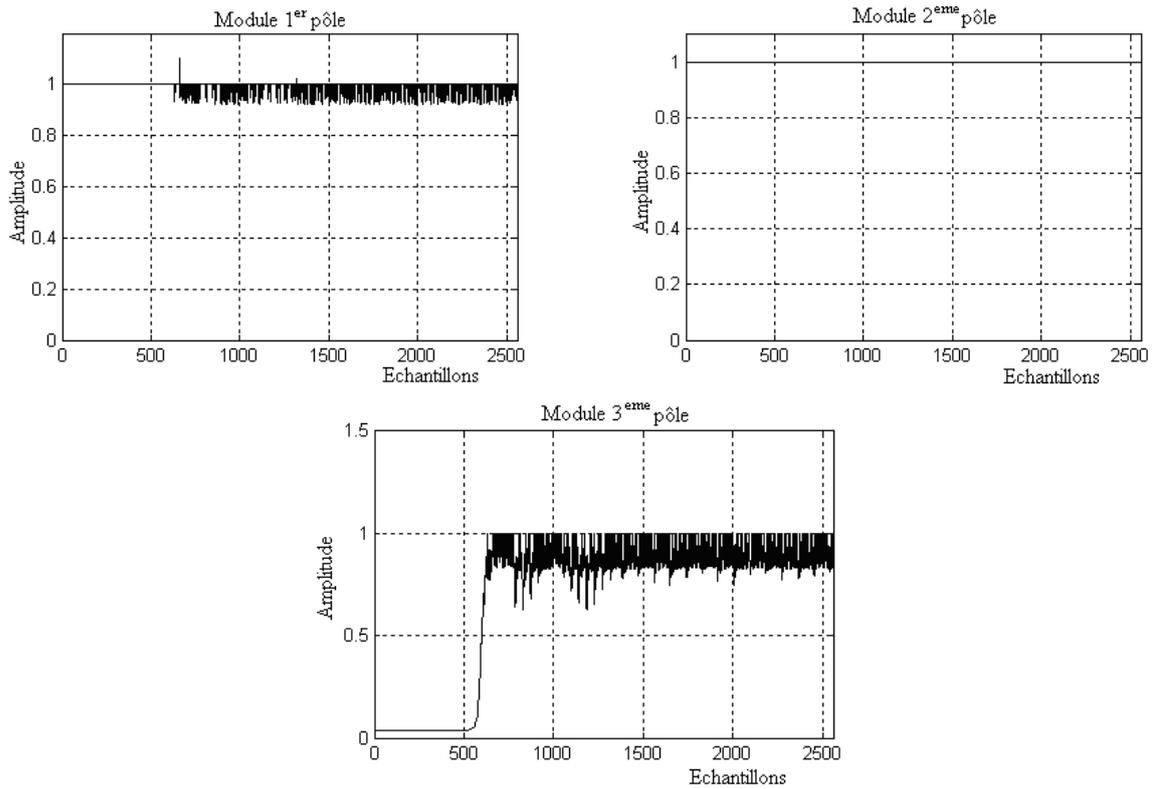


Fig.III.13b: Evolution du module des pôles du prédicteurs aller ($L=3, k_0$ non entier)

III.4.2.2 Cas de plusieurs sinusoides pures

Pour un signal parfaitement prédictible x_n donnée par :

$$x_n = \sum_{i=1}^m a_i \sin\left(\frac{2\pi}{k_i} n\right) \tag{III.63}$$

modélisable par un ordre AR(M), où $M=2m$ pour $m=1, 2, 3$.

Pour un filtre d'ordre L et un k_i entier ou non, nous avons observé les oscillations des variables de l'algorithme pour les cas : $L=AR(M)$, $L < AR(M)$ sous-estimé, et $L > AR(M)$ surestimé.

D'après les résultats du tableau III.3, on constate que l'algorithme est stable pour un ordre $AR(M=2m)$ égal à l'ordre L , et instable pour un ordre $AR(M=2m)$ supérieur à l'ordre L (ordre AR sous-estimé). Mais, pour l'ordre $AR(M=2m)$ inférieur à l'ordre L (ordre AR surestimé), on remarque que l'algorithme est stable, si la fréquence d'échantillonnage n'est pas un multiple entier d'au moins l'une des fréquences de la sinusoides c'est-à-dire lorsque l'un des k_i est non entier. Sinon l'algorithme diverge.

x_n	Ordre du prédicteur (L)							
	2	3	4	5	6	7	8	
1 sinusoides								
k_0 entier	✓	✗	✗	✗	✗	✗	✗	
k_0 non entier	✓	✓	✓	✓	✓	✓	✓	
2 sinusoides								
k_1 entier								
k_2 entier	✗	✗	✓	✗	✗	✗	✗	
k_1 entier								
k_2 non entier	✗	✗	✓	✓	✓	✓	✓	
k_1 non entier								
k_2 non entier	✗	✗	✓	✓	✓	✓	✓	
3 sinusoides								
k_1 entier								
k_2 entier	✗	✗	✗	✗	✓	✗	✗	
k_3 entier								
k_1 entier								
k_2 entier	✗	✗	✗	✗	✓	✓	✓	
k_3 non entier								
k_1 entier								
k_2 non entier	✗	✗	✗	✗	✓	✓	✓	
k_3 non entier								
k_1 non entier								
k_2 non entier	✗	✗	✗	✗	✓	✓	✓	
k_3 non entier								

Tableau III.5: Comportement des algorithmes FRLS pour un ordre L par rapport à $AR(M=2m)$ et k_i

$$F_e = k_i F_i, \quad k_i : \text{entier ou non entier}, m=1 ; 2 ; 3$$

F_e : Fréquence d'échantillonnage, F_i : Fréquence de la sinusoides

✓ : Convergence de l'algorithme, ✗ : Divergence de l'algorithme

III.4.2.3 Influence d'un bruit additif

Reprenons le cas critique où le signal d'entrée x_n représente une sinusoïde dont la fréquence d'échantillonnage F_e est multiple entier de la fréquence de la sinusoïde F_0 et où l'ordre du modèle AR est surestimé. Ajoutons un niveau de bruit au signal d'entrée. On peut alors écrire :

$$x_{u_n} = x_n + u_n \quad (\text{III.64})$$

où u_n est un bruit blanc gaussien stationnaire de variance σ_u^2 .

Les figures III.14a et III.14b correspondent au cas : AR(2), $L=3$ (surestimé) et k_0 entier. Le signal d'entrée est une sinusoïde bruitée de fréquence $F_0 = 2kHz$, la fréquence d'échantillonnage est $F_e = 16kHz$ et dont le rapport signal sur bruit (SNR_{dB}) est de 60 dB.

La figure III.14a illustre la stabilité de cet algorithme, où l'indicateur de divergence reste très faible et la variable de divergence fluctue autour de sa valeur optimale. Les pôles du prédicteur sortent parfois du cercle unité (Fig. III.14b), mais aucune divergence n'a été détectée.

L'ajout du bruit à l'entrée n'a pas fait de changement pour les cas AR(M)= L et AR(M) > L (sous-estimé), l'algorithme FRLS reste respectivement stable et instable. Mais pour le cas AR(M) < L (surestimé), l'algorithme est stable, quelle que soit la fréquence d'échantillonnage par rapport à la fréquence de la sinusoïde

Si les signaux sont suffisamment bruités, ce qui est le cas en pratique, il n'y aura pas de problème d'instabilité numérique, car les variances sont alors limitées par la variance du bruit perturbateur et les pôles des prédicteurs se trouvent à l'intérieur du cercle unité.

Un facteur d'oubli très proche de l'unité stabilise l'algorithme pour ce type de signaux, mais ces valeurs stabilisatrices avec comme signal d'entrée des sinusoïdes pures sont visiblement grandes devant la taille des prédicteurs. Dans cette situation, l'algorithme FRLS va perdre toutes ses performances en vitesse de convergence, et on va se retrouver en présence d'un algorithme sans oubli exponentiel. Pour cette raison et en vertu de la dynamique des signaux utilisés, une technique simple de régularisation avec une composante de bruit très faible s'avère très efficace, afin de maintenir les pôles à l'intérieur du cercle unité.

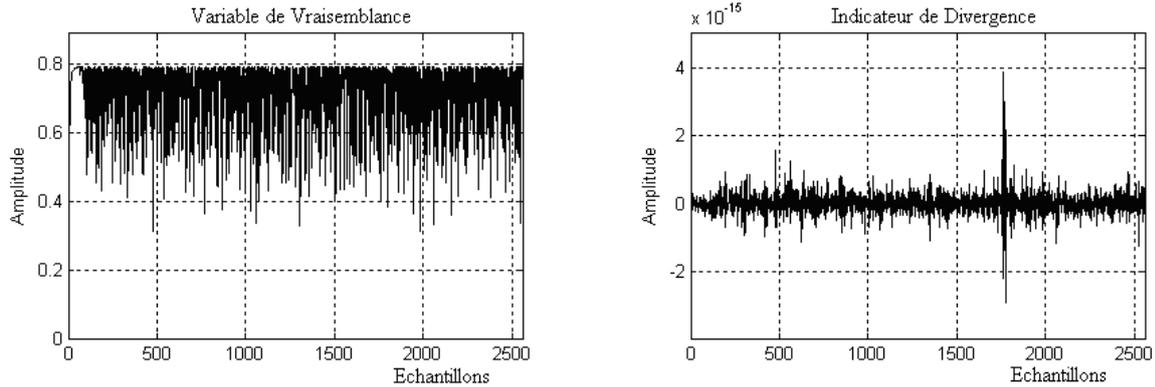


Fig.III.14a: Evolution des variables γ_n et ξ_n ($L=3$, k_0 entier) avec $SNR_{dB}=60dB$

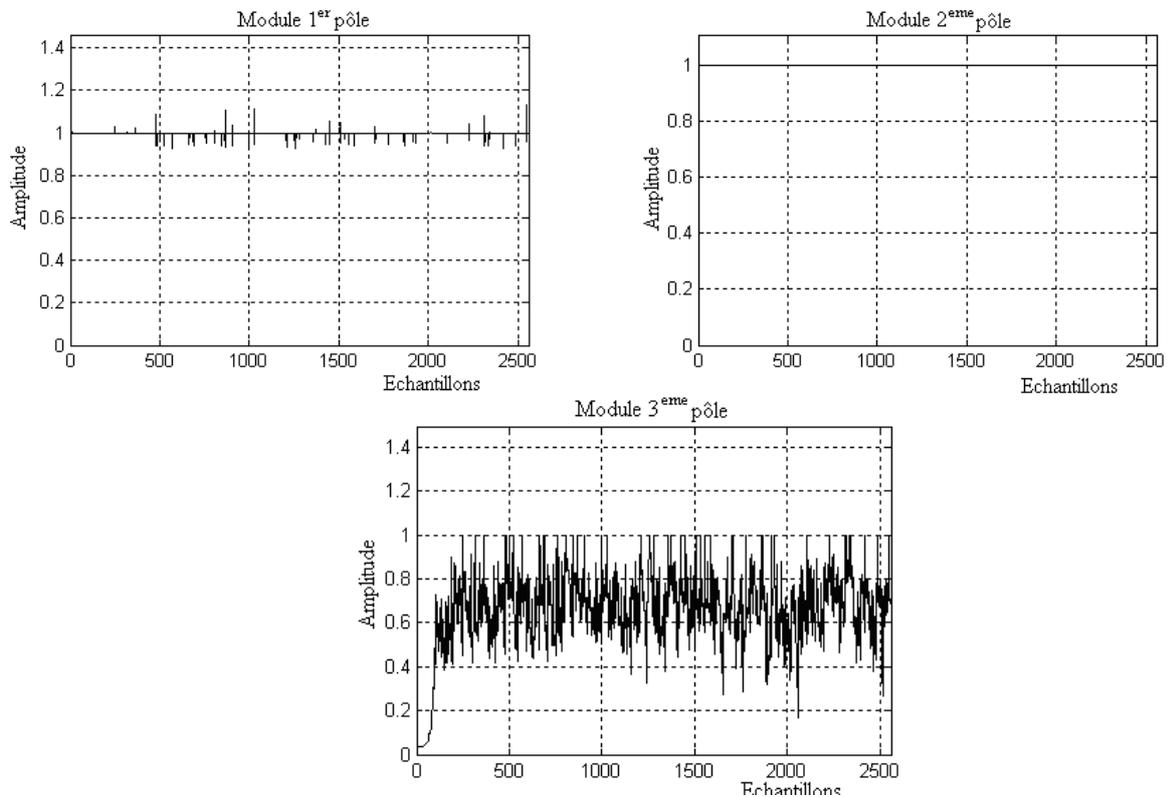


Fig.III.14b: Evolution du module des pôles du prédicteurs aller ($L=3$, k_0 entier) avec $SNR_{dB}=60dB$

III.4.3 Comportement des algorithmes FRLS avec des signaux non stationnaire

Nous n'avons jamais observé de problèmes de stabilité numérique pour les algorithmes NS-FRLS proposés quand le signal d'entrée est stationnaire. Mais pour un signal d'entrée non stationnaire comme la parole, des instabilités numériques ont été observées pour la partie prédiction. En effet, la stabilisation numérique des algorithmes FRLS est basée sur l'hypothèse que le signal d'entrée excite continuellement et elle n'est valable que dans un domaine restreint de λ . Lorsque le signal d'entrée est un signal de parole (zones voisées très prédictible et zone de

silence), l'algorithme RLS risque de diverger à cause du mauvais conditionnement de la matrice d'autocorrélation à court terme. Dans le cas des algorithmes NS-FRLS, cela se répercute par des variances d'erreurs de prédictions très faibles et la chute à zéro de la variable de vraisemblance. Pour résoudre ce problème, nous pouvons mettre en œuvre des procédures de réinitialisation. Dans [47], l'auteur régularise la variance de l'erreur de prédiction aller en y ajoutant une constante positive faible alors que d'autres [59, 60] réinitialisent l'algorithme en détectant la chute à 0 de la variable de vraisemblance. Une autre technique [61] qui consiste à profiter de l'intervalle de silence pour bloquer l'adaptation et de la reprendre dès qu'il y a signal utile et cela en estimant en permanence l'énergie du signal de parole. Ces problèmes disparaissent en prenant un facteur d'oubli supérieur à la borne inférieure de sa stabilité numérique mais on perd du point de vue de la vitesse de convergence et de la capacité de poursuite de l'algorithme.

La limitation du facteur d'oubli à la condition (III.35), c'est-à-dire $(1-1/2L) < \lambda < 1$ due à l'instabilité numérique de l'algorithme FRLS, empêche l'utilisation de facteurs d'oubli faibles ce qui confère à l'algorithme une capacité de poursuite faible. Du fait de la stabilité numérique de la partie filtrage des algorithmes RLS, nous pouvons accélérer ce dernier afin d'améliorer sa capacité de poursuite [62], l'idée est de multiplier le gain par un pas scalaire μ_n variable dans le temps :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu_n \mathbf{g}_n \bar{\varepsilon}_n \quad (\text{III.65})$$

En exprimant l'erreur a posteriori en fonction de l'erreur a priori, il vient :

$$\varepsilon_n = (1 - \mu_n \mathbf{g}_n^T \mathbf{x}_n) \bar{\varepsilon}_n \quad (\text{III.66})$$

En choisissant

$$\mu_n = \frac{1}{\mathbf{g}_n^T \mathbf{x}_n} = \frac{1}{1 - \gamma_n} \quad (\text{III.67})$$

L'erreur a posteriori est nulle. Pour éviter de possible perturbation de l'algorithme ($\gamma_n \approx 1$), nous pouvons introduire un facteur μ_A , appelé facteur d'accélération tel que

$$\mu_n = \frac{1}{1 - \mu_A \gamma_n} \quad (\text{III.68})$$

La partie filtrage de cet algorithme appelé RLS accéléré (ARLS) s'écrit :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\gamma_n}{1 - \mu_A \gamma_n} \tilde{\mathbf{k}}_n \bar{\varepsilon}_n \quad (\text{III.69})$$

Avec cette mise à jour, le facteur d'oubli équivalent pour la partie filtrage, en régime asymptotique, est donné par :

$$\lambda_{\text{eq}} = \frac{(1 - \mu_A)\lambda}{1 - \mu_A \lambda} \quad (\text{III.70})$$

où $\lambda_{\text{eq}} < \lambda$ lorsque $0 < \mu_A < 1$. Notons que nous retrouvons la partie filtrage de l'algorithme RLS classique pour $\mu_A = 0$. Nous donnons les courbes des $MSE(n)$ pour l'algorithme NS-FRLS accéléré pour différentes valeurs de μ_A . Le signal d'entrée x_n est un bruit corrélé stationnaire avec un spectre équivalent au spectre moyen de la parole, appelé bruit USASI dans le domaine de l'annulation d'écho acoustique. Ce signal, de moyenne nulle et de variance égale à 0.32, est échantillonné à 16 kHz. Le signal désiré d_n incorpore une variation du système (gain linéaire variable) comme montré dans Fig.III.15. Nous avons un ordre $L=256$, un facteur d'oubli $\lambda = (1 - 1/3L) = 0.9987$, $E_0 = 1$ pour $\mu_A = 0$ et $\mu_A = 0.99$. La Fig.III.16 montre l'amélioration apportée par le facteur d'accélération μ_A pour la capacité de poursuite. μ_A

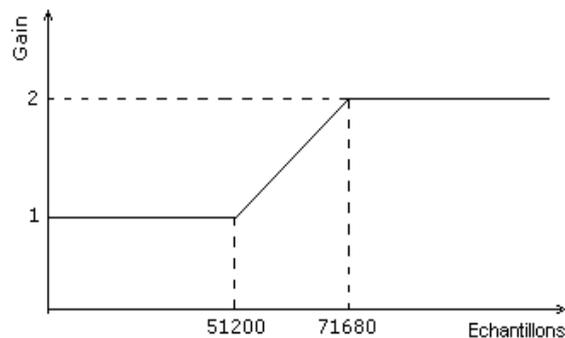


Fig.III.15 : Gain variable

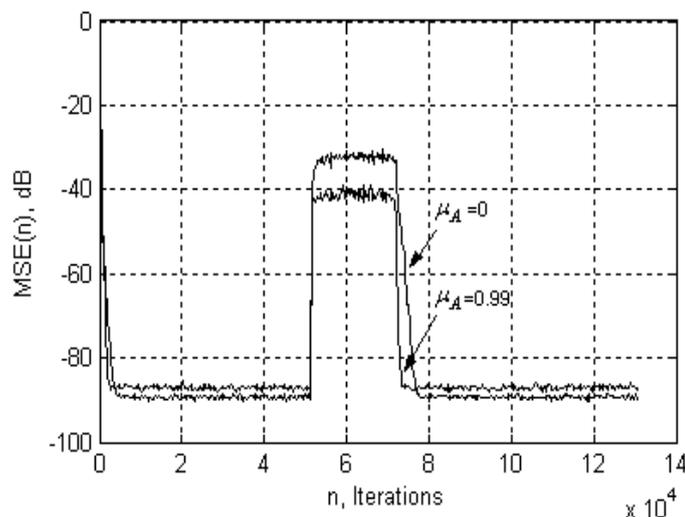


Fig.III.16 : Comparaison de la capacité de poursuite des algorithmes NS-FRLS et NS-FRLS accéléré

III.5 Conclusion

Nous avons analysé les propriétés numériques de l'algorithme FRLS stabilisé en utilisant un modèle du premier ordre de la propagation des erreurs numériques. Nous avons présenté de nouvelles versions numériquement stables des algorithmes des moindres carrés récursifs rapides (NS-FRLS). La condition de stabilité s'est avérée capable de maintenir une bonne performance de convergence par des simulations sur ordinateur. Ces algorithmes ainsi modifiés sont stables numériquement pour un choix convenable des paramètres (μ^a, μ^b, μ^c) , μ_s et du facteur d'oubli λ . Ces nouvelles méthodes augmentent faiblement la complexité de calcul des algorithmes initiaux de L multiplications supplémentaire échantillon traité pour chaque méthode. L'introduction de la variable $NGE(n)$, nous a permis de comparer et montrer l'évolution de la stabilité des algorithmes. Nous avons observé que les résultats sont en accord avec les expressions théoriques. La stabilité numérique a été vérifiée par simulation sur de très grandes durées de signal x_n stationnaire et pour différents ordres de filtre.

Par la suite, nous avons présenté une étude sur le comportement des algorithmes des moindres carrés récursifs rapides numériquement stable (NS-FRLS) avec des sinusoides pures. D'une part, nous avons rassemblé un certain nombre de résultats sur la position des pôles et la divergence de l'algorithme. Pour stabiliser l'algorithme, nous avons proposé un choix de la fréquence d'échantillonnage par rapport à la fréquence de la sinusoides. D'autre part, nous montrons ici la possibilité d'avoir des algorithmes stables en fonctionnement continu avec un bruit additif. Si le bruit est nul, le signal d'entrée est un signal prédictible et les coefficients de prédiction sont tels que les pôles soient sur le cercle unité. Au fur et à mesure que le niveau du bruit augmente, les coefficients varient de sorte que les pôles soient ramenés vers l'intérieur du cercle. Des techniques de réinitialisation peuvent être utilisées sur ses algorithmes lorsque le signal d'entrée est de la parole. Nous avons montré aussi que nous pouvons accéléré l'algorithme en utilisant une méthode qui permet d'avoir la borne inférieure de stabilité pour le facteur d'oubli et donc d'améliorer sa capacité de poursuite tandis que sa partie prédiction reste stable.

Chapitre IV

ALGORITHME DES MOINDRES CARRES RAPIDES SIMPLIFIE ET MODIFIE (M-SMFTF)

IV.1 Introduction

Les algorithmes utilisés dans le système d'annulation d'écho acoustique représentent une charge de calcul trop importante qu'il faut chercher à réduire. Notre but est alors de développer de nouveaux algorithmes plus efficaces, appréciés en terme de compromis entre la qualité du système d'annulation d'écho acoustique et la complexité de calcul engendré. Pour cela, une étude théorique a porté sur les algorithmes d'adaptation utilisés dans le système d'annulation d'écho acoustique pour nous fournir une bonne connaissance du système et nous permettre de proposer quelques améliorations entraînant une réduction satisfaisante de calcul. Nous proposons ainsi un nouvel algorithme rapide (M-SMFTF) pour le filtrage adaptatif. Cet algorithme présente un intérêt certain, pour l'adaptation de filtres très longs comme ceux utilisés dans les problèmes d'annulation d'écho acoustique, à cause de sa complexité réduite, sa stabilité numérique et sa convergence en présence du signal de parole. La complexité de calcul de cet algorithme est de $6L$ et celle-ci peut être considérablement réduite à $(2L+4P)$ quand on utilise un prédicteur aller réduit d'ordre P ($P \ll L$). L'algorithme M-SMFTF surpasse les algorithmes adaptatifs classiques en raison de sa vitesse de convergence qui approche celle de l'algorithme RLS et de sa complexité de calcul qui est légèrement plus grande que celle de l'algorithme NLMS. Dans ce qui suit, nous décrivons les différentes étapes pour l'obtention du nouvel algorithme M-SMFTF. Une réduction de la taille du prédicteur aller de l'algorithme M-SMFTF est proposé. Par la suite les propriétés numériques de l'algorithme M-SMFTF sont analysées. Ces analyses montrent la supériorité en termes de capacité de poursuite de l'algorithme proposé à l'algorithme NS-FRLS.

IV.2 Algorithmes adaptatifs

En général le problème d'identification de système implique la construction d'une évaluation d'un système inconnu avec seulement deux signaux, un signal d'entrée x_n et un signal désiré d_n . Typiquement le système inconnu est modélisé linéairement avec une réponse impulsionnelle finie (FIR) $\mathbf{w}_{\text{opt},L}$ de taille L , et des algorithmes adaptatifs de filtrage sont utilisés pour converger itérativement vers une estimation de la réponse [30, 63]. Le système est estimé par le filtre adaptatif $\mathbf{w}_{L,n}$, avec une sortie à ce dernier $\hat{y}_n = \mathbf{w}_{L,n-1}^T \mathbf{x}_{L,n}$, où le signal d'erreur en sortie est (Fig. IV.1):

$$\bar{\varepsilon}_{L,n} = d_n - \hat{y}_n \quad (\text{IV.1})$$

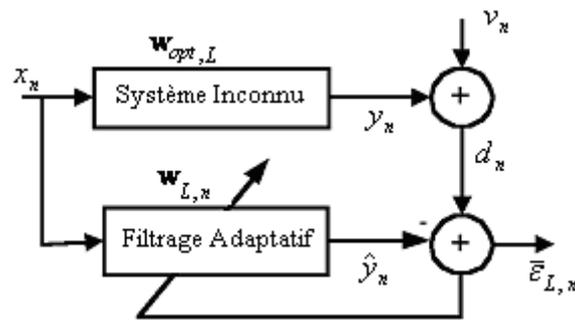


Fig. IV.1: Système de filtrage FIR adaptatif

Le filtre est mis à jour à chaque instant par la rétroaction de l'erreur d'estimation proportionnelle au gain d'adaptation $\mathbf{g}_{L,n}$:

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \mathbf{g}_{L,n} \bar{\varepsilon}_{L,n} \quad (\text{IV.2})$$

Les différents algorithmes sont distingués par le calcul de ce gain.

Il y a deux classes principales d'algorithmes adaptatifs. La première classe est l'algorithme des moindres carrés moyens (LMS), qui est basée sur la méthode du gradient stochastique. L'algorithme LMS demeure l'algorithme adaptatif le plus largement répandu dans un grand nombre d'applications pratiques. L'algorithme du gradient stochastique normalisé NLMS, de complexité de calcul $2L$, est une variante du LMS dont le gain d'adaptation est normalisé par l'énergie du signal d'entrée. Il a été intensivement étudié, et beaucoup de résultats théoriques sur ses performances en régime permanent ont été obtenus [64]. L'autre classe est l'algorithme des moindres carrés récursifs (RLS) qui minimise une somme déterministe des carrés des erreurs

[65]. L'algorithme RLS résout ce problème, mais aux dépens de la plus grande complexité de calcul qui est de $O(L^2)$. Un grand nombre d'algorithmes RLS rapides (FRLS) ont été développés au cours de ces dernières années, mais malheureusement plus l'algorithme FRLS, en terme d'efficacité de calcul, est meilleur plus ses problèmes liés à la stabilité numérique sont importants [65, 66]. La meilleure complexité de calcul $O(L)$ réalisée par ces algorithmes rapides est une conséquence directe de la structure d'invariance par décalage du vecteur de signal d'entrée. Les versions rapides les plus connues de ces algorithmes, à savoir, Kalman rapide (FK : Fast Kalman) [33], FAEST (Fast A-posteriori Error Sequential Technique) [34], et FTF (Fast Transversal Filter) [35], sont dérivées du RLS en mettant à jour le vecteur de gain de Kalman, qui est employé pour mettre à jour le filtre adaptatif, par l'introduction des prédicteurs aller et retour.

Une manière alternative de réduire la complexité de calcul de l'algorithme RLS est décrite dans [67]. En se servant de l'interpolation de RLS aussi bien que la prédiction, l'algorithme produit des coefficients du filtre transverse sans souffrir de l'instabilité numérique qui est attribuée à l'algorithme FTF. Le gain de Kalman est produit à chaque itération en termes des résiduels d'interpolation. Les résiduels d'interpolation sont calculés pour un ordre L d'une manière récursive. L'algorithme dérivé, appelé estimateur de gain de Kalman (KaGE : Kalman Gain Estimator). La complexité de calcul est $(L \log_2(L))$ opérations par itération, ceci est réalisé par l'intermédiaire d'une approche « diviser-et-conquérir ». Cette complexité de calcul demeure grande comparée aux algorithmes FRLS. Cependant, tous les algorithmes FRLS développés sont numériquement instables une fois mis en application dans l'arithmétique de précision finie; les erreurs d'approximation s'accumulent et mènent à l'instabilité numérique. Ces problèmes numériques ont été étudiés intensivement dans la littérature [50-52, 68-78]. La plupart de ces méthodes emploient quelques formules redondantes, disponibles dans les algorithmes FRLS, pour estimer les erreurs numériques et rétroagir ces dernières à un point quelconque de l'algorithme afin de modifier ses propriétés numériques. La stabilité numérique peut également être obtenue en insérant des facteurs de fuite dans les prédicteurs aller et retour [79]. La complexité de calcul est de $8L$ à $11L$.

Une autre manière de réduire la complexité de calcul, des algorithmes FRLS, a été proposée dans [80, 81]. Quand le signal d'entrée peut être exactement modélisé par un prédicteur d'ordre P ($P \ll L$), l'algorithme du filtre transverse de Newton rapide (FNFTF : Fast Newton Transversal Filter) évite le fonctionnement des prédicteurs aller et retour d'ordre L , ce qui serait exigé par

l'algorithme FRLS. Les quantités exigées sont extrapolées par les prédicteurs d'ordre P . Ainsi, la complexité de l'algorithme FNTF chute et devient $(2L+12P)$ multiplications au lieu de $8L$. L'algorithme FNTF, se trouve ainsi entre les algorithmes de LMS et de RLS en termes de coût de calcul et de convergence.

Une plus grande réduction de complexité de calcul dans la partie prédiction de l'algorithme FNTF peut être réalisée en adaptant les prédicteurs aller et retour avec deux algorithmes réduits de la taille LMS [82] ou en employant l'algorithme basé sur la prédiction retour des moindres carrés rapides (FBPLS : Fast Backward Predictor Least Squares) [83]. Les complexités de calculs de ces deux derniers algorithmes sont $(2L+6P)$ et $(2L+5P+1.5P^2)$ respectivement.

Dans [84], une approche alternative a été introduite pour réduire la complexité de calcul de ces algorithmes, qui fait réduire la fréquence d'adaptation du filtre. La mise à jour du filtre est alors dite sous-échantillonnée avec un rapport de sous-échantillonnage M donnée. Deux stratégies apparaissent: La première consiste à faire un traitement par blocs en résolvant les équations normales à chaque M échantillons, c'est l'algorithme BRLS (Block RLS) [85]. La deuxième alternative consiste à utiliser la même stratégie que le RLS en calculant le nouveau filtre et d'autres quantités auxiliaires à partir des données disponibles M échantillons auparavant. Cette démarche amène à l'algorithme SU RLS (Subsampled Updating RLS). Cependant, celui-ci présente des problèmes d'instabilité dus aux erreurs d'arrondi. Une version rapide et stable du SU RLS (FSU SFTF) [86] en utilisant la notion de structure de déplacement ainsi que la technique de convolution rapide par utilisation de la transformée de Fourier rapide (FFT). En fait, la stratégie du BRLS s'avère plus intéressante pour les grandes valeurs de M (comparables à L) alors que le FSU SFTF est plus intéressant pour des valeurs intermédiaires. La même technique est utilisée pour l'algorithme FNTF, c'est le cas de l'algorithme FSU FNTF [87]. Le prix à payer est cependant, l'introduction d'un retard de $O(M)$ échantillons dans le traitement.

Récemment, l'algorithme simplifié de type FTF (SMFTF) [88] développé pour l'usage de l'annulation d'écho acoustique. Cet algorithme dérivé de l'algorithme FTF où le gain d'adaptation est obtenu seulement à partir des variables de la prédiction aller. Les variables de la prédiction retour, qui sont la source principale d'instabilité numérique dans les algorithmes de FRLS, sont complètement enlevées. Un développement théorique important [71] montre, que les variables retour sont théoriquement redondantes. La complexité de calcul de l'algorithme SMFTF est de $7L$. Il est moins complexe que l'algorithme numériquement stable NS-FRLS qui est de $8L$.

IV.3 Réduction de Complexité

IV.3.1 Proposition de l'algorithme M-SMFTF

Nous proposons un nouvel algorithme rapide simplifié et modifié (M-SMFTF) pour le filtrage adaptatif. Cet algorithme présente un intérêt certain, pour l'adaptation de filtres très longs comme ceux utilisés dans les problèmes d'annulation d'écho acoustique, à cause de sa complexité réduite, sa stabilité numérique et sa convergence en présence du signal de parole. L'algorithme simplifié de type FTF (SMFTF) [88] qui est dérivé de l'algorithme FTF où le gain d'adaptation est obtenu seulement à partir des variables de la prédiction aller. Les variables de la prédiction retour, qui sont la source principale d'instabilité numérique dans les algorithmes de FRLS [50-52], sont complètement enlevées. En utilisant seulement les variables de la prédiction aller et en ajoutant une petite constante de régulation et un facteur de fuite.

Le gain d'adaptation, donné par $\mathbf{g}_{L,n} = \gamma_{L,n} \tilde{\mathbf{k}}_{L,n}$, où la variable de vraisemblance $\gamma_{L,n}$ est calculée en utilisant directement la définition, elle est donnée par :

$$\gamma_{L,n} = \frac{1}{1 + \tilde{\mathbf{k}}_{L,n}^T \mathbf{x}_{L,n}} \quad (\text{IV.3})$$

Cet algorithme SMFTF de complexité $7L$, présente les mêmes performances que les algorithmes FRLS.

L'algorithme M-SMFTF que nous proposons utilise les mêmes étapes que l'algorithme SMFTF mais calcule d'une autre façon la variable de vraisemblance.

L'égalité des deux expressions, (I.53a) et (I.53b) du chapitre I, du gain de Kalman dual étendu à l'ordre $(L+1)$ nous donne, l'expression du gain de Kalman dual de l'algorithme FRLS:

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{L,n}}{\lambda \alpha_{L,n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} - \frac{\bar{r}_{L,n}}{\lambda \beta_{L,n-1}} \begin{bmatrix} -\mathbf{b}_{L,n-1} \\ 1 \end{bmatrix} \quad (\text{IV.4})$$

et si nous retirons complètement la partie prédiction de l'algorithme, ce cas correspond théoriquement au signal d'entrée bruit blanc ($\mathbf{a}_{L,n} = \mathbf{b}_{L,n} = \mathbf{0}_L$), alors le gain de Kalman dual devient:

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ * \end{bmatrix} = \begin{bmatrix} \frac{x_n}{\lambda \alpha_{L,n-1}} \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} \quad (\text{IV.5})$$

où (*) indique une composante inutilisée. Après les L premières itérations, une version décalée de la première composante est propagée à toutes les autres composantes de $\tilde{\mathbf{k}}_{L,n}$:

$$\tilde{\mathbf{k}}_{L,n} = \left[\frac{x_n}{\lambda\alpha_{L,n-1}}, \frac{x_{n-1}}{\lambda\alpha_{L,n-2}}, \dots, \frac{x_{n-L+1}}{\lambda\alpha_{L,n-L}} \right]^T \quad (\text{IV.6})$$

En régime permanent, pour un signal d'entrée stationnaire, la variance des erreurs de prédiction aller peut être rapproché par une quantité proportionnelle à la puissance du signal d'entrée. Alors les termes $\lambda\alpha_{L,n-i}$, convergent vers une valeur proportionnelle à la puissance du signal d'entrée et nous pouvons approximativement écrire :

$$\mathbf{g}_{L,n} = \gamma_{L,n} \tilde{\mathbf{k}}_{L,n} \approx \kappa \frac{\mathbf{x}_{L,n}}{\sigma_x^2} \quad (\text{IV.7})$$

où κ est une constante positive. Ce gain d'adaptation a une forme semblable au gain d'adaptation des algorithmes NLMS. Considérons le gain de Kalman dual (IV.4) mis à jour avec les deux prédicteurs aller et retour, nous pouvons voir que les composantes les plus significatives, les dernières, du prédicteur retour affectent les derniers termes du gain de Kalman dual et cette contribution n'est pas transmise à d'autres positions inférieures de $\tilde{\mathbf{k}}_{L,n}$ en raison de la propriété de rétrogradation dans le calcul du gain de Kalman dual. Ceci signifie d'une part que les composantes les plus significatives du filtre adaptatif $\mathbf{w}_{L,n}$ sont mises à jour en employant les variables récursives de la prédiction aller, et d'autre part que les variables récursives de la prédiction retour affectent les dernières composantes $\mathbf{w}_{L,n}$ dont lesquelles, dans la plupart des applications pratiques, ont des valeurs très petites. Cette observation suggère d'écarter complètement le prédicteur retour de l'algorithme FTF. Ecarter le prédicteur retour ne signifie pas que les dernières composantes $\mathbf{w}_{L,n}$ ne sont pas mises à jour, mais elles sont mises à jour par des composantes provenant des positions inférieures de $\tilde{\mathbf{k}}_{L,n}$. En plus de ces observations, un développement théorique important [71] montre, que les variables retour sont théoriquement redondantes, données par les variables de la prédiction aller et le gain de Kalman dual. En éliminant les variables inutiles, on obtient un système minimal dont la propagation des erreurs est stable pour un signal d'entrée constamment excité.

Nous écartons toutes les variables de la prédiction retour de (IV.4) et utilisons seulement les variables aller pour calculer le gain de Kalman dual:

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ * \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} \quad (\text{IV.8a})$$

$$\mathbf{a}_{L,n} = \mathbf{a}_{L,n-1} + \bar{e}_{L,n} \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \quad (\text{IV.8b})$$

Cet algorithme n'est pas très robuste avec un signal d'entrée non stationnaire comme le signal de parole. La première difficulté vient de $\alpha_{L,n} = \lambda\alpha_{L,n-1} \rightarrow 0$ durant les périodes de silence du signal. Cette convergence vers zéro met les algorithmes FRLS et leurs versions numériquement stables dans des situations très difficiles. L'instabilité peut se produire puisque nous essayons d'exécuter des divisions numériques par des valeurs très petites. Pour se préserver contre cette possibilité, comme c'est souvent fait avec l'algorithme NLMS, nous additionnons une petite constante positive c_a au dénominateur

$$\frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1}} \rightarrow \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1} + c_a} \quad (\text{IV.9})$$

La deuxième difficulté est que le prédicteur aller est verrouillé sur ses dernières valeurs durant ces mêmes périodes de silence du signal d'entrée. Les algorithmes FRLS ont été développés dans le cas de pré-fenêtrage où tous les vecteurs sont initialisés à zéro de sorte que l'algorithme commence à s'adapter. Dans ces conditions, quand le signal d'entrée disparaît et réapparaît après une longue période, l'algorithme peut diverger en raison de ces valeurs non nulles du prédicteur. En d'autres termes, l'algorithme n'est pas bien initialisé quand le signal réapparaît. Dans de telles conditions, il est préférable que le prédicteur aller $\mathbf{a}_{L,n}$ revienne de nouveau à zéro. Pour cela, on applique l'opération suivante:

$$\mathbf{a}_{L,n} \rightarrow \eta \mathbf{a}_{L,n} \quad (\text{IV.10})$$

où η est une constante très proche de 1 appelé facteur de fuite [71].

Nous proposons de réduire davantage la complexité de calcul en employant une nouvelle méthode récursive pour calculer la variable de vraisemblance [89]. Pour cela, remplaçons la quantité (*), qui n'a pas été employé dans l'expression (IV.8a), par la variable $c_{L,n}$, nous obtenons:

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ c_{L,n} \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1} + c_a} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} \quad (\text{IV.11})$$

En exploitant certaines propriétés d'invariance par décalage du vecteur signal d'entrée étendu à l'ordre $(L+1)$, nous pouvons écrire le vecteur d'entrée selon deux formes différentes :

$$\mathbf{x}_{L+1,n} = [\mathbf{x}_{L,n}^T \quad x_{n-L}]^T \quad (\text{IV.12a})$$

$$\mathbf{x}_{L+1,n} = [x_n \quad \mathbf{x}_{L,n-1}^T]^T \quad (\text{IV.12b})$$

En multipliant à gauche, les membres de gauche et de droite de l'expression (IV.11) par les expressions (IV.12a) et (IV.12b) respectivement, on obtient l'égalité suivante :

$$\mathbf{x}_{L,n}^T \tilde{\mathbf{k}}_{L,n} + c_{L,n} x_{n-L} = \mathbf{x}_{L,n-1}^T \tilde{\mathbf{k}}_{L,n-1} + \frac{\bar{e}_{L,n}^2}{\lambda \alpha_{L,n-1} + c_a} \quad (\text{IV.13})$$

L'expression (IV.13) peut être réécrite comme suit:

$$(1 + \mathbf{x}_{L,n}^T \tilde{\mathbf{k}}_{L,n}) = (1 + \mathbf{x}_{L,n-1}^T \tilde{\mathbf{k}}_{L,n-1}) + \delta_{L,n} \quad (\text{IV.14})$$

où

$$\delta_{L,n} = \frac{\bar{e}_{L,n}^2}{\lambda \alpha_{L,n-1} + c_a} - c_{L,n} x_{n-L} \quad (\text{IV.15})$$

En prenant l'inverse de l'expression (IV.14), nous obtenons:

$$\frac{1}{(1 + \mathbf{x}_{L,n}^T \tilde{\mathbf{k}}_{L,n})} = \frac{1}{(1 + \mathbf{x}_{L,n-1}^T \tilde{\mathbf{k}}_{L,n-1}) + \delta_{L,n}} \quad (\text{IV.16})$$

En manipulant la relation (IV.16), nous obtenons une nouvelle formule récursive pour calculer la variable de vraisemblance, elle est donnée comme suit:

$$\gamma_{L,n} = \frac{\gamma_{L,n-1}}{1 + \delta_{L,n} \gamma_{L,n-1}} \quad (\text{IV.17})$$

L'algorithme M-SMFTF, de complexité de calcul $6L$, est listé dans la Table IV.1.

Initialisation: $\mathbf{w}_{L,0} = \mathbf{a}_{L,0} = \tilde{\mathbf{k}}_{L,0} = \mathbf{0}_L$; $\gamma_{L,0} = 1$; $\alpha_{L,0} = \lambda^L E_0$; $E_0 \geq \sigma_x^2 \frac{L}{100}$

Variables disponibles à l'instant n : $\mathbf{a}_{L,n-1}$; $\tilde{\mathbf{k}}_{L,n-1}$; $\gamma_{L,n-1}$; $\alpha_{L,n-1}$; $\mathbf{w}_{L,n-1}$

Nouvelles informations: x_n , d_n ;

- Partie Prédiction:

$$\bar{e}_{L,n} = x_n - \mathbf{a}_{L,n-1}^T \mathbf{x}_{L,n-1};$$

$$\alpha_{L,n} = \lambda \alpha_{L,n-1} + \gamma_{L,n-1} \bar{e}_{L,n}^2$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ c_{L,n} \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{L,n}}{\lambda \alpha_{L,n-1} + c_a} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix};$$

$$\mathbf{a}_{L,n} = \eta \left\{ \mathbf{a}_{L,n-1} + \bar{e}_{L,n} \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \right\};$$

$$\delta_{L,n} = \frac{\bar{e}_{L,n}^2}{\lambda \alpha_{L,n-1} + c_a} - c_{L,n} x_{n-L}; \quad \gamma_{L,n} = \frac{\gamma_{L,n-1}}{1 + \delta_{L,n} \gamma_{L,n-1}}$$

- Partie Filtrage:

$$\bar{e}_{L,n} = d_n - \mathbf{w}_{L,n-1}^T \mathbf{x}_{L,n};$$

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \bar{e}_{L,n} \gamma_{L,n} \tilde{\mathbf{k}}_{L,n}.$$

Table IV.1: Résumé de l'algorithme M-SMFTF ($6L$)

IV.3.2 Proposition de l'algorithme RM-SMFTF

Dans les algorithmes FRLS (RLS rapide), l'ordre de prédiction du signal d'excitation dépend de la taille L du système à identifier, et non de la statistique de ce même signal. Or, pour pallier à cet inconvénient une classe d'algorithmes FNTF (Fast Newton Transversal Filter) [80], introduit un degré de liberté supplémentaire quant au choix de l'ordre de prédiction. Dans ces algorithmes, il est supposé que le processus observé est autorégressif d'ordre P , $AR(P)$. Par une méthode d'extrapolation, la matrice d'autocorrélation d'ordre L se construit à partir d'une estimation de la matrice d'autocorrélation d'ordre P . Dans ce cas, il n'est plus nécessaire de propager des vecteurs de prédiction d'ordre L dans la partie prédiction de l'algorithme FRLS puisque dans ces vecteurs, P coefficients seulement sont non nuls. Les réductions de la taille des prédicteurs des algorithmes FTF [35] ont été employées avec succès dans les algorithmes FNTF [80, 81]. Nous pouvons alors, facilement réduire la taille de la partie prédiction de l'algorithme M-SMFTF proposé [90]. En effet, si nous désignons par P l'ordre du prédicteur et par L la taille du filtre adaptatif, l'algorithme M-SMFTF à prédicteur réduit (RM-SMFTF) [89, 90] va comporter les 3 phases suivantes :

- Prédiction d'ordre P à l'instant n .
- Extrapolation pour calculer le gain de Kalman dual et la variable de vraisemblance.
- Filtrage et Adaptation.

Le prédicteur aller et le gain de Kalman dual sont donnés respectivement par:

$$\mathbf{a}_{L,n} = \begin{bmatrix} \mathbf{a}_{P,n} \\ \mathbf{0}_{L-P} \end{bmatrix} \quad (\text{IV.18})$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ c_{L,n} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{P,n}}{\lambda\alpha_{P,n-1} + c_a} \begin{bmatrix} 1 \\ -\mathbf{a}_{P,n-1} \\ \mathbf{0}_{L-P} \end{bmatrix} \quad (\text{IV.19})$$

où P est beaucoup plus petit que L ($P \ll L$) et où $\mathbf{0}_{L-P}$ est un vecteur nul d'ordre $(L-P)$. Les $(P+1)$ premières composantes de $\tilde{\mathbf{k}}_{L,n}$ sont mises à jour en utilisant les variables aller de taille réduites, les dernières composantes sont justes une version décalée de la $(P+1)^{\text{ième}}$ composante de $\tilde{\mathbf{k}}_{L,n}$. Pour cet algorithme, nous avons besoin de deux variables de vraisemblance.

La première variable de vraisemblance $\gamma_{P,n}$, est employé pour mettre à jour la variance des erreurs de prédiction aller $\alpha_{P,n}$:

$$\gamma_{P,n} = \frac{\gamma_{P,n-1}}{1 + \delta_{P,n} \gamma_{P,n-1}} \quad (\text{IV.20a})$$

avec

$$\delta_{P,n} = \frac{\bar{e}_{P,n}^2}{\lambda \alpha_{P,n-1} + c_a} - c_{P,n} x_{n-P} \quad (\text{IV.20b})$$

où $c_{P,n}$ est la $(P+1)$ ^{ième} composante de $\tilde{\mathbf{k}}_{L,n}$.

La seconde variable de vraisemblance $\gamma_{L,n}$, est employé pour mettre à jour le prédicteur aller $\mathbf{a}_{P,n}$ d'ordre P et le filtre transverse $\mathbf{w}_{L,n}$:

$$\gamma_{L,n} = \frac{\gamma_{L,n-1}}{1 + \delta_{L,n} \gamma_{L,n-1}} \quad (\text{IV.21a})$$

avec

$$\delta_{L,n} = \frac{\bar{e}_{P,n}^2}{\lambda \alpha_{P,n-1} + c_a} - c_{L,n} x_{n-L} \quad (\text{IV.21b})$$

L'algorithme RM-SMFTF, de complexité de calcul $(2L+4P)$, est listé dans la Table IV.2.

Initialisation: $\mathbf{w}_{L,0} = \tilde{\mathbf{k}}_{L,0} = \mathbf{0}_L$; $\gamma_{L,0} = 1$; $\mathbf{a}_{P,0} = \mathbf{0}_P$; $\gamma_{P,0} = 1$; $\alpha_{P,0} = \lambda^P E_0$; $E_0 \geq \sigma_x^2 \frac{P}{100}$

Variables disponibles à l'instant n : $\mathbf{a}_{P,n-1}$; $\gamma_{P,n-1}$; $\alpha_{P,n-1}$; $\gamma_{L,n-1}$; $\tilde{\mathbf{k}}_{L,n-1}$; $\mathbf{w}_{L,n-1}$

Nouvelles informations: x_n , d_n ;

- Partie Prédiction:

$$\bar{e}_{P,n} = x_n - \mathbf{a}_{P,n-1}^T \mathbf{x}_{P,n-1};$$

$$\alpha_{P,n} = \lambda \alpha_{P,n-1} + \gamma_{P,n-1} \bar{e}_{P,n}^2;$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ c_{L,n} \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} + \frac{\bar{e}_{P,n}}{\lambda \alpha_{P,n-1} + c_a} \begin{bmatrix} 1 \\ -\mathbf{a}_{P,n-1} \\ 0_{L-P} \end{bmatrix};$$

$$\tilde{\mathbf{k}}_{P,n-1} = \tilde{\mathbf{k}}_{L,n-1} (1:P);$$

$$\mathbf{a}_{P,n} = \eta \left\{ \mathbf{a}_{P,n-1} + \bar{e}_{P,n} \gamma_{L,n-1} \tilde{\mathbf{k}}_{P,n-1} \right\};$$

$$c_{P,n} = \tilde{\mathbf{k}}_{L,n} (P+1);$$

$$\delta_{P,n} = \frac{\bar{e}_{P,n}^2}{\lambda \alpha_{P,n-1} + c_a} - c_{P,n} x_{n-P}; \quad \gamma_{P,n} = \frac{\gamma_{P,n-1}}{1 + \delta_{P,n} \gamma_{P,n-1}};$$

$$\delta_{L,n} = \frac{\bar{e}_{P,n}^2}{\lambda \alpha_{P,n-1} + c_a} - c_{L,n} x_{n-L}; \quad \gamma_{L,n} = \frac{\gamma_{L,n-1}}{1 + \delta_{L,n} \gamma_{L,n-1}};$$

- Partie Filtrage:

$$\bar{e}_{L,n} = d_n - \mathbf{w}_{L,n-1}^T \mathbf{x}_{L,n};$$

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \bar{e}_{L,n} \gamma_{L,n} \tilde{\mathbf{k}}_{L,n}.$$

Table IV.2: Résumé de l'algorithme RM-SMFTF $(2L+4P; P \ll L)$

IV.4 Analyse des erreurs de propagation de l'algorithme M-SMFTF

IV.4.1 Analyse des erreurs numériques de la partie prédiction

Nous étudions la propagation des erreurs de toutes les quantités récursives de la partie prédiction de l'algorithme M-SMFTF. Supposons que les erreurs numériques sont petites, le modèle de propagation d'erreur dans les variables récursives peut être rapproché par le modèle linéaire suivant:

$$\Delta \boldsymbol{\varphi}_n = \mathbf{F}(n) \Delta \boldsymbol{\varphi}_{n-1} \quad (\text{IV.22})$$

où $\Delta \boldsymbol{\varphi}_n$ représente le vecteur d'état des erreurs à l'instant n que nous pouvons écrire alors comme suit :

$$\Delta \boldsymbol{\varphi}_n = \begin{bmatrix} \Delta \boldsymbol{\varphi}_n^{aT} & \Delta \boldsymbol{\varphi}_n^{kT} \end{bmatrix}^T \quad (\text{IV.23})$$

où

$$\Delta \boldsymbol{\varphi}_n^a = \begin{bmatrix} \Delta \mathbf{a}_{L,n} \\ \Delta \alpha_{L,n} \end{bmatrix} \text{ et } \Delta \boldsymbol{\varphi}_n^k = \begin{bmatrix} \Delta \tilde{\mathbf{k}}_{L,n} \\ \Delta \gamma_{L,n} \end{bmatrix} \quad (\text{IV.24})$$

désignent respectivement les erreurs cumulées jusqu'à l'instant n dans les variables récursives aller et Kalman. La matrice $\mathbf{F}(n)$ de dimension $(2L+2) \times (2L+2)$:

$$\mathbf{F}(n) = \begin{bmatrix} \mathbf{F}_{11}(n) & \mathbf{F}_{12}(n) \\ \mathbf{F}_{21}(n) & \mathbf{F}_{22}(n) \end{bmatrix} \quad (\text{IV.25})$$

représente la matrice de transition.

Le système (IV.22) est dit stable, en régime permanent et au sens moyen, si toutes les valeurs propres de $E\{\mathbf{F}(n)\}$ sont de module inférieur à 1 [68].

Nous différencions $\mathbf{a}_{L,n}$ et $\alpha_{L,n}$ pour obtenir les erreurs numériques dans les variables aller:

$$\Delta \mathbf{a}_{L,n} = \eta \Delta \mathbf{a}_{L,n-1} + \eta \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \Delta \bar{e}_{L,n} + \eta \bar{e}_{L,n} \tilde{\mathbf{k}}_{L,n-1} \Delta \gamma_{L,n-1} + \eta \bar{e}_{L,n} \gamma_{L,n-1} \Delta \tilde{\mathbf{k}}_{L,n-1} \quad (\text{IV.26a})$$

$$\Delta \alpha_{L,n} = \lambda \Delta \alpha_{L,n-1} + 2\gamma_{L,n-1} \bar{e}_{L,n} \Delta \bar{e}_{L,n} + \bar{e}_{L,n}^2 \Delta \gamma_{L,n-1} \quad (\text{IV.26b})$$

où

$$\Delta \bar{e}_{L,n} = -\mathbf{x}_{L,n-1}^T \Delta \mathbf{a}_{L,n-1} \quad (\text{IV.27})$$

En regroupant les expressions (IV.26a) et (IV.26b), sous une forme plus condensée, il vient

$$\Delta \boldsymbol{\varphi}_n^a = \begin{bmatrix} \eta (\mathbf{I}_L - \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \mathbf{x}_{L,n-1}^T) & 0_L \\ -2\bar{e}_{L,n} \gamma_{L,n-1} \mathbf{x}_{L,n-1}^T & \lambda \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^a + \begin{bmatrix} \eta \bar{e}_{L,n} \gamma_{L,n-1} \mathbf{I}_L & \eta \bar{e}_{L,n} \tilde{\mathbf{k}}_{L,n-1} \\ 0_L^T & \bar{e}_{L,n}^2 \end{bmatrix} \Delta \boldsymbol{\varphi}_{n-1}^k \quad (\text{IV.28})$$

où 0_L et \mathbf{I}_L indiquent respectivement un vecteur nul de dimension L et une matrice identité d'ordre L .

De la même manière, nous différencions $\tilde{\mathbf{k}}_{L,n}$ et $\gamma_{L,n}$ pour obtenir les erreurs numériques dans les variables de Kalman. En différenciant l'expression (IV.11), nous obtenons:

$$\begin{aligned} \begin{bmatrix} \Delta\tilde{\mathbf{k}}_{L,n} \\ \Delta c_{L,n} \end{bmatrix} &= \begin{bmatrix} 0 \\ \Delta\tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} - \frac{\lambda \bar{e}_{L,n}}{(\lambda \alpha_{L,n-1} + c_a)^2} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} \Delta\alpha_{L,n-1} \\ &\quad - \frac{1}{\lambda \alpha_{L,n-1} + c_a} \left\{ \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} \mathbf{x}_{L,n-1}^T + \bar{e}_{L,n} \begin{bmatrix} \mathbf{0}_L^T \\ \mathbf{I}_L \end{bmatrix} \right\} \Delta\mathbf{a}_{L,n-1} \end{aligned} \quad (\text{IV.29})$$

cette dernière s'écrit sous forme plus condensée :

$$\begin{bmatrix} \Delta\tilde{\mathbf{k}}_{L,n} \\ \Delta c_{L,n} \end{bmatrix} = \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\varphi}_{n-1}^a + \begin{bmatrix} \mathbf{0}_L^T \\ \mathbf{I}_L \end{bmatrix} \Delta\tilde{\mathbf{k}}_{L,n-1} \quad (\text{IV.30})$$

où

$$\mathbf{Q}_{L+1}(n) = \frac{1}{\lambda \alpha_{L,n-1} + c_a} \begin{bmatrix} -\mathbf{x}_{L,n-1}^T & -\frac{\lambda \bar{e}_{L,n}}{\lambda \alpha_{L,n-1} + c_a} \\ \mathbf{a}_{L,n-1} \mathbf{x}_{L,n-1}^T - \bar{e}_{L,n} \mathbf{I}_L & \mathbf{a}_{L,n-1} \frac{\lambda \bar{e}_{L,n}}{\lambda \alpha_{L,n-1} + c_a} \end{bmatrix} \quad (\text{IV.31})$$

représente une matrice $(L+1) \times (L+1)$. Nous pouvons déduire de (IV.30):

$$\Delta\tilde{\mathbf{k}}_{L,n} = [\mathbf{I}_L \quad \mathbf{0}_L] \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\varphi}_{n-1}^a + [\mathbf{I}_L \quad \mathbf{0}_L] \begin{bmatrix} \mathbf{0}_L^T \\ \mathbf{I}_L \end{bmatrix} \Delta\tilde{\mathbf{k}}_{L,n-1} \quad (\text{IV.32})$$

et

$$\Delta c_{L,n} = \mathbf{q}_{L+1}^T \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\varphi}_{n-1}^a + \mathbf{q}_L^T \Delta\tilde{\mathbf{k}}_{L,n-1} \quad (\text{IV.33})$$

où le vecteur $\mathbf{q}_{L+1} = [0, 0, \dots, 0, 1]^T$ permet d'extraire la $(L+1)^{\text{ième}}$ composante d'un vecteur d'ordre $L+1$.

Les erreurs numériques dans le calcul de la variable de vraisemblance sont :

$$\Delta\gamma_{L,n} = \frac{\gamma_{L,n}}{\gamma_{L,n-1}} (1 - \delta_{L,n} \gamma_{L,n}) \Delta\gamma_{L,n-1} - \gamma_{L,n}^2 \Delta\delta_{L,n} \quad (\text{IV.34})$$

avec

$$\Delta\delta_{L,n} = \bar{e}_{L,n} \Delta\tilde{k}_{1,n} + \tilde{k}_{1,n} \Delta\bar{e}_{L,n} - x_{n-L} \Delta c_{L,n} \quad (\text{IV.35})$$

où l'élément $\tilde{k}_{1,n}$ représente la première composante de $\tilde{\mathbf{k}}_{L,n}$ que nous pouvons écrire sous la forme $\tilde{k}_{1,n} = \bar{\mathbf{q}}_L^T \tilde{\mathbf{k}}_{L,n}$, où le vecteur $\bar{\mathbf{q}}_L = [1, 0, \dots, 0]^T$ permet d'extraire la première composante d'un vecteur d'ordre L . Les erreurs numériques dans le calcul de $\tilde{k}_{1,n}$ sont :

$$\Delta\tilde{k}_{1,n} = \bar{\mathbf{q}}_L^T \mathbf{Q}_{L+1}(n) \Delta\boldsymbol{\varphi}_{n-1}^a \quad (\text{IV.36})$$

En regroupant les relations (IV.32) et (IV.34), les erreurs numériques dans les variables de Kalman sont données par:

$$\Delta\boldsymbol{\Phi}_n^k = \begin{bmatrix} [\mathbf{I}_L \ 0_L] \mathbf{Q}_{L+1}(n) \\ (\mathbf{u}_L^T(n), u(n)) \end{bmatrix} \Delta\boldsymbol{\Phi}_{n-1}^k + \begin{bmatrix} [\mathbf{I}_L \ 0_L] \begin{bmatrix} 0_L^T \\ \mathbf{I}_L \end{bmatrix} & 0_L \\ x_{n-L} \gamma_{L,n}^2 \mathbf{q}_L^T & c^\gamma(n) \end{bmatrix} \Delta\boldsymbol{\Phi}_{n-1}^k \quad (\text{IV.37})$$

où

$$\mathbf{u}_L^T(n) = \frac{2\gamma_{L,n}^2 \bar{e}_{L,n}}{\lambda\alpha_{L,n-1} + c_a} \mathbf{x}_{L,n-1}^T + \frac{\gamma_{L,n}^2 x_{n-L}}{\lambda\alpha_{L,n-1} + c_a} (\mathbf{q}_L^T \mathbf{a}_{L,n-1} \mathbf{x}_{n-1}^T - \bar{e}_{L,n} \mathbf{q}_L^T \mathbf{I}_L) \quad (\text{IV.38a})$$

$$u(n) = \frac{\lambda\gamma_{L,n}^2 \bar{e}_{L,n}}{(\lambda\alpha_{L,n-1} + c_a)^2} (\bar{e}_{L,n} + x_{n-L} \mathbf{q}_L^T \mathbf{a}_{L,n-1}) \quad (\text{IV.38b})$$

$$c^\gamma(n) = \frac{\gamma_{L,n}}{\gamma_{L,n-1}} (1 - \delta_{L,n} \gamma_{L,n}) \quad (\text{IV.38c})$$

En manipulant les équations (IV.25) et (IV.26), nous pouvons écrire:

$$(1 - \delta_{L,n} \gamma_{L,n}) = \frac{1}{1 + \delta_{L,n} \gamma_{L,n-1}} \quad (\text{IV.39})$$

où

$$\delta_{L,n} \gamma_{L,n-1} = \frac{\gamma_{L,n-1} \bar{e}_{L,n}^2}{\lambda\alpha_{L,n-1} + c_a} - \gamma_{L,n-1} c_{L,n} x_{n-L} \quad (\text{IV.40})$$

Nous pouvons également écrire $x_{n-L} = \mathbf{x}_{L,n-1}^T \mathbf{q}_L$ et déduire $c_{L,n}$ de (IV.11) :

$$c_{L,n} = \mathbf{q}_L^T \tilde{\mathbf{k}}_{L,n-1} - \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1} + c_a} \mathbf{q}_L^T \mathbf{a}_{L,n-1} \quad (\text{IV.41})$$

Ainsi, l'expression (IV.40) devient:

$$\delta_{L,n} \gamma_{L,n-1} = \frac{\gamma_{L,n-1} \bar{e}_{L,n}^2}{\lambda\alpha_{L,n-1} + c_a} - \mathbf{q}_L^T \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \mathbf{x}_{L,n-1}^T \mathbf{q}_L + \frac{\gamma_{L,n-1} \bar{e}_{L,n}}{\lambda\alpha_{L,n-1} + c_a} \mathbf{q}_L^T \mathbf{a}_{L,n-1} \mathbf{x}_{L,n-1}^T \mathbf{q}_L \quad (\text{IV.42})$$

Nous pouvons déduire les matrices $\mathbf{F}_{ij}(n)$ ($i=1, 2; j=1, 2$) des expressions (IV.28) et (IV.37).

Nous réalisons un modèle qui fait une analyse moyenne de propagation des erreurs en déterminant, en régime permanent et en supposant que x_n est un signal stationnaire de moyenne nulle, la valeur moyenne de $\mathbf{F}(n)$. En outre, nous supposons également que toutes les variables de l'algorithme ont atteint leurs véritables valeurs asymptotiques. Remplaçons $\gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1}$ par $\mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1}$ et utilisons la récurrence $\mathbf{R}_{L,n} = \lambda \mathbf{R}_{L,n-1} + \mathbf{x}_{L,n} \mathbf{x}_{L,n}^T$ à l'instant $(n-1)$, nous obtenons

$\mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T = \mathbf{R}_{L,n-1} - \lambda \mathbf{R}_{L,n-2}$, alors l'expression $\gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \mathbf{x}_{L,n-1}^T$ devient $(\mathbf{I}_L - \lambda \mathbf{R}_{L,n-1}^{-1} \mathbf{R}_{L,n-2})$.

En mode asymptotique, nous pouvons écrire:

$$\mathbf{q}_L^T \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \mathbf{x}_{L,n-1}^T \mathbf{q}_L \rightarrow (1-\lambda) \quad (\text{IV.43})$$

$$\eta(\mathbf{I}_L - \gamma_{L,n-1} \tilde{\mathbf{k}}_{L,n-1} \mathbf{x}_{L,n-1}^T) = \eta \lambda \mathbf{R}_{L,n-1}^{-1} \mathbf{R}_{L,n-2} \rightarrow \eta \lambda \mathbf{I}_L \quad (\text{IV.44})$$

$$\frac{\gamma_{L,n}}{\gamma_{L,n-1}} \rightarrow 1 \quad (\text{IV.45})$$

$$\gamma_{L,n-1} \bar{e}_{L,n}^2 = \alpha_{L,n} - \lambda \alpha_{L,n-1} \rightarrow (1-\lambda) \alpha_F \quad (\text{IV.46})$$

où α_F est la valeur asymptotique de la variance des erreurs de la prédiction aller. En divisant l'expression (IV.46) par $(\lambda \alpha_{L,n-1} + c_a)$ et sous la condition suivante $(c_a / \lambda \alpha_F) \ll 1$ nous obtenons:

$$\frac{\gamma_{L,n-1} \bar{e}_{L,n}^2}{\lambda \alpha_{L,n-1} + c_a} \rightarrow (\lambda^{-1} - 1) \quad (\text{IV.47})$$

En utilisant la valeur asymptotique,

$$(\lambda \alpha_{L,n-1} + c_a) \rightarrow \frac{\lambda \gamma_F \mathbb{E}\{\bar{e}_{L,n}^2\}}{1-\lambda} + c_a \quad (\text{IV.48})$$

où γ_F est la valeur asymptotique de la variable $\gamma_{L,n-1}$. Ainsi la matrice $\mathbf{F}_{21}(n)$ peut être vu comme une matrice proportionnelle à $(1-\lambda)$. En prenant l'espérance mathématique de la matrice de transition $\mathbb{E}\{\mathbf{F}(n)\}$ et le cas limite $\lambda \rightarrow 1$, et en supposant $\mathbb{E}\{\bar{e}_{L,n}\} = 0$, nous obtenons:

$$\mathbb{E}\{\mathbf{F}_{11}(n)\} = \begin{bmatrix} \eta \lambda \mathbf{I}_L & \mathbf{0}_L \\ \mathbf{0}_L^T & \lambda \end{bmatrix} \quad (\text{IV.49a}); \quad \mathbb{E}\{\mathbf{F}_{12}(n)\} = \begin{bmatrix} \mathbf{0}_{L \times L} & \mathbf{0}_L \\ \mathbf{0}_L^T & \mathbb{E}\{\bar{e}_{L,n}^2\} \end{bmatrix} \quad (\text{IV.49b})$$

$$\mathbb{E}\{\mathbf{F}_{21}(n)\} = \mathbf{0}_{(L+1) \times (L+1)} \quad (\text{IV.50a}); \quad \mathbb{E}\{\mathbf{F}_{22}(n)\} = \begin{bmatrix} \mathbf{M}^k & \mathbf{0}_L \\ \mathbf{0}_L^T & \mathbb{E}\{c^\gamma(n)\} \end{bmatrix} \quad (\text{IV.50b})$$

avec

$$\mathbf{M}^k = \begin{bmatrix} \mathbf{0}_{L-1}^T & \mathbf{0} \\ \mathbf{I}_{L-1} & \mathbf{0}_{L-1} \end{bmatrix} \quad (\text{IV.51a}); \quad \mathbb{E}\{c^\gamma(n)\} = \frac{1}{\lambda^{-1} + \lambda - 1} \quad (\text{IV.51b})$$

Dans ce cas-ci, la matrice $\mathbb{E}\{\mathbf{F}(n)\}$ est très semblable à une matrice triangulaire supérieure par blocs; nous pouvons donc écrire $\mathbb{E}\{\mathbf{F}(n)\}$ comme suit:

$$\mathbb{E}\{\mathbf{F}(n)\} = \begin{bmatrix} \mathbb{E}\{\mathbf{F}_{11}(n)\} & \mathbb{E}\{\mathbf{F}_{12}(n)\} \\ \mathbf{0}_{(L+1) \times (L+1)} & \mathbb{E}\{\mathbf{F}_{22}(n)\} \end{bmatrix} \quad (\text{IV.52})$$

Nous pouvons montrer aussi que, pour les valeurs λ et η inférieures à 1, les matrices $E\{\mathbf{F}_{11}(n)\}$ et $E\{\mathbf{F}_{22}(n)\}$ ont le module de leurs valeurs propres inférieur à 1. En conséquence, le système est numériquement stable en moyenne, pour λ et η compris entre 0 et 1 [90].

Nous pouvons approcher les erreurs numériques dans le calcul des variable aller et des variables de Kalman par des modèles du premier ordre déduites respectivement de (IV.28) et (IV.37) :

$$\Delta \mathbf{a}_{L,n} = \eta (\mathbf{I}_L - \mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T) \Delta \mathbf{a}_{L,n-1} + \mathbf{p}_a(n) \quad (\text{IV.53a})$$

$$\Delta \alpha_{L,n} = \lambda \Delta \alpha_{L,n-1} + p_\alpha(n) \quad (\text{IV.53b})$$

$$\Delta \tilde{\mathbf{k}}_{L,n} = \mathbf{M}^k \Delta \tilde{\mathbf{k}}_{L,n-1} + \mathbf{p}_k(n) \quad (\text{IV.54a})$$

$$\Delta \gamma_{L,n} = c^\gamma(n) \Delta \gamma_{L,n-1} + p_\gamma(n) \quad (\text{IV.54b})$$

où $\mathbf{p}_a(n)$ et $p_\alpha(n)$ sont des fonctions qui ne dépendent pas des erreurs $\Delta \mathbf{a}_{L,n-1}$ et $\Delta \alpha_{L,n-1}$ respectivement, de même $\mathbf{p}_k(n)$ et $p_\gamma(n)$ sont des fonctions qui ne dépendent pas des erreurs $\Delta \tilde{\mathbf{k}}_{L,n-1}$ et $\Delta \gamma_{L,n-1}$ respectivement.

L'analyse employant le comportement moyen du système (IV.22) ne permet pas de trouver la véritable condition de stabilité, parce que les erreurs numériques peuvent être de moyenne nulle mais de variance non bornée. Calculons la matrice de covariance des erreurs numériques dans le calcul du prédicteur aller $\Delta \mathbf{a}_{L,n}$ (IV.53a), pour cela nous utilisons l'approche statistique $E\{\Delta \mathbf{a}_{L,n} \Delta \mathbf{a}_{L,n}^T\} = \mathbf{A}_{L,n}$. En supposant l'indépendance entre les variables $\Delta \mathbf{a}_{L,n-1}$ et $\mathbf{p}_a(n)$, nous pouvons écrire:

$$\begin{aligned} \mathbf{A}_{L,n} = & \eta^2 \mathbf{A}_{L,n-1} - \eta^2 \mathbf{A}_{L,n-1} E\left\{ \left(\mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T \right)^T \right\} - \eta^2 E\left\{ \mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T \right\} \mathbf{A}_{L,n-1} \\ & + \eta^2 E\left\{ \mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T \Delta \mathbf{a}_{L,n-1} \Delta \mathbf{a}_{L,n-1}^T \left(\mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T \right)^T \right\} + E\left\{ \mathbf{p}_a(n) \mathbf{p}_a^T(n) \right\} \end{aligned} \quad (\text{IV.55})$$

Nous notons que la quantité $\mathbf{R}_{L,n}$ varie lentement comparé au signal d'entrée x_n , et nous supposons que les composantes du vecteur $\Delta \mathbf{a}_{L,n}$ sont indépendantes entre elles et indépendantes des différentes variables théoriques de l'algorithme.

En outre, le signal d'entrée est une séquence de variables aléatoires gaussiennes décorrélée de variance σ_x^2 , ce qui permet d'écrire $E\{\mathbf{R}_{L,n-1}^{-1}\} = (1-\lambda)\mathbf{R}_{L,xx}^{-1}$ où $\mathbf{R}_{L,xx} = E\{\mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T\} = \sigma_x^2 \mathbf{I}_L$, et $E\{\mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n-1} \mathbf{x}_{L,n-1}^T\}$ a une valeur asymptotique de $(1-\lambda)\mathbf{I}_L$ [68]. L'expression (IV.55) devient:

$$\mathbf{A}_{L,n} = (\eta^2 - 2\eta^2(1-\lambda))\mathbf{A}_{L,n-1} + \eta^2(1-\lambda)^2 \sigma_x^{-4} \mathbf{H}(n) + \mathbf{E}\{\mathbf{p}_a(n)\mathbf{p}_a^T(n)\} \quad (\text{IV.56})$$

où

$$\mathbf{H}(n) = \mathbf{E}\{\mathbf{x}_{L,n-1}\mathbf{x}_{L,n-1}^T \Delta \mathbf{a}_{l,n-1} \Delta \mathbf{a}_{l,n-1}^T \mathbf{x}_{L,n-1}\mathbf{x}_{L,n-1}^T\} \quad (\text{IV.57})$$

est une matrice carré d'ordre L , dont les éléments s'écrivent :

$$h_{ij} = \sum_{k=1}^L \sum_{m=1}^L \mathbf{E}\{\Delta a_{k,n-1} \Delta a_{m,n-1} x_{n-k} x_{n-m} x_{n-i} x_{n-j}\} \quad (\text{IV.58})$$

En développant l'expression (IV.58), nous trouvons:

$$h_{ij} = \sum_{k=1}^L [\sigma_{\Delta a,n-1}^2 \mathbf{E}\{x_{n-k}^2\} \mathbf{E}\{x_{n-i} x_{n-j}\} + \sigma_{\Delta a,n-1}^2 \mathbf{E}\{x_{n-k} x_{n-i}\} \mathbf{E}\{x_{n-i} x_{n-j}\} + \sigma_{\Delta a,n-1}^2 \mathbf{E}\{x_{n-k} x_{n-j}\} \mathbf{E}\{x_{n-i} x_{n-j}\}] \quad (\text{IV.59})$$

où $\sigma_{\Delta a,n-1}^2 = \mathbf{E}\{\Delta a_{l,n-1}^2\}$. Si $i \neq j \Rightarrow h_{ij} = 0$; et si $i = j \Rightarrow$

$$h_{ii} = \sigma_{\Delta a,n-1}^2 \sigma_x^2 \sum_{k=1}^L [\mathbf{E}\{x_{n-k}^2\} + 2\mathbf{E}\{x_{n-k} x_{n-i}\}] \quad (\text{IV.60})$$

Nous obtenons:

$$h_{ij} = \begin{cases} \sigma_{\Delta a,n-1}^2 \sigma_x^4 (L+2) & i = j \\ 0 & i \neq j \end{cases} \quad (\text{IV.61})$$

En conclusion, nous obtenons l'expression suivante:

$$\mathbf{A}_{L,n} = G \mathbf{A}_{L,n-1} + \mathbf{E}\{\mathbf{p}_a(n)\mathbf{p}_a^T(n)\} \quad (\text{IV.62})$$

où

$$G = \eta^2 (1 - 2(1-\lambda) + (1-\lambda)^2 (L+2)) \quad (\text{IV.63})$$

En supposant que la quantité $\mathbf{E}\{\mathbf{p}_a(n)\mathbf{p}_a^T(n)\}$ est bornée, la condition de stabilité de l'équation (IV.62) est donnée par la solution de l'inégalité suivante :

$$|G| < 1 \quad (\text{IV.64})$$

En appliquant la condition (IV.64), nous obtenons [91]:

$$\lambda > 1 - \frac{1 + \sqrt{1 + \left(\frac{1}{\eta^2} - 1\right)(L+2)}}{(L+2)} \quad (\text{IV.65})$$

Nous notons que la limite inférieure de cette condition est toujours plus petite que la limite inférieure de la condition (III.36) de l'algorithme numériquement stable de FRLS, ce qui signifie que nous pouvons choisir de plus petites valeurs pour le facteur d'oubli pour l'algorithme

proposé et par conséquent, on a une vitesse de convergence plus rapide et une meilleure capacité de poursuite.

IV.4.2 Analyse de convergence

Nous définissons le vecteur d'erreur des coefficients du filtre adaptatif à l'instant n par [54]:

$$\Delta \mathbf{w}_{L,n} = \mathbf{w}_{\text{opt},L} - \mathbf{w}_{L,n} \quad (\text{IV.66})$$

L'erreur de sortie $\bar{e}_{L,n}$ s'écrit alors:

$$\bar{e}_{L,n} = v_n + \mathbf{x}_{L,n}^T \Delta \mathbf{w}_{L,n-1} \quad (\text{IV.67})$$

La récurrence $\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \mathbf{g}_{L,n} \bar{e}_{L,n}$ sur le vecteur d'erreur des coefficients $\Delta \mathbf{w}_{L,n}$ donne:

$$\Delta \mathbf{w}_{L,n} = [\mathbf{I}_L - \gamma_{L,n} \tilde{\mathbf{k}}_{L,n} \mathbf{x}_{L,n}^T] \Delta \mathbf{w}_{L,n-1} - \gamma_{L,n} \tilde{\mathbf{k}}_{L,n} v_n \quad (\text{IV.68})$$

Dans le cas d'un bruit blanc en entrée ($\mathbf{a}_{L,n} = \mathbf{0}_L$, $\bar{e}_{L,n} = x_n$) et en supposant $c_a = 0$, l'expression (IV.11) peut être écrite :

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L,n} \\ c_{L,n} \end{bmatrix} = \begin{bmatrix} x_n \\ \lambda \alpha_{L,n-1} \\ \tilde{\mathbf{k}}_{L,n-1} \end{bmatrix} \quad (\text{IV.69})$$

Après les L premières itérations, une version décalée de la première composante est propagée à toutes les autres composantes de $\tilde{\mathbf{k}}_{L,n}$:

$$\tilde{\mathbf{k}}_{L,n} = \left[\frac{x_n}{\lambda \alpha_{L,n-1}}, \frac{x_{n-1}}{\lambda \alpha_{L,n-2}}, \dots, \frac{x_{n-L+1}}{\lambda \alpha_{L,n-L}} \right]^T \quad (\text{IV.70})$$

En régime asymptotique, et utilisons l'expression récurrente de la variance des erreurs de prédiction aller $\alpha_{L,n} = \lambda \alpha_{L,n-1} + \gamma_{L,n-1} \bar{e}_{L,n}^2$, nous pouvons écrire:

$$\alpha_{L,n-i} \rightarrow \alpha(\infty) = \gamma(\infty) \frac{\sigma_x^2}{1-\lambda} \quad (\text{IV.71})$$

où $\gamma(\infty)$ est la valeur asymptotique de la variable de vraisemblance $\gamma_{L,n}$.

Le gain de Kalman dual devient:

$$\tilde{\mathbf{k}}_{L,n} = \frac{1}{\lambda \alpha(\infty)} \mathbf{x}_{L,n} \quad (\text{IV.72})$$

Replaçons la variable $\gamma_{L,n}$ par sa valeur en régime permanent $\gamma(\infty)$ et utilisons (IV.71) et (IV.72), l'expression récurrente (IV.68) devient:

$$\Delta \mathbf{w}_{L,n} = \left[\mathbf{I}_L - \frac{(1-\lambda)}{\lambda \sigma_x^2} \mathbf{x}_{L,n} \mathbf{x}_{L,n}^T \right] \Delta \mathbf{w}_{L,n-1} - \frac{(1-\lambda)}{\lambda \sigma_x^2} v_n \quad (\text{IV.73})$$

En prenant l'espérance des deux termes de la relation (IV.73) et en exploitant l'hypothèse d'indépendance, nous trouvons:

$$E\{\Delta \mathbf{w}_{L,n}\} = (2 - \lambda^{-1}) E\{\Delta \mathbf{w}_{L,n-1}\} \quad (\text{IV.74})$$

En régime permanent, lorsque le filtre a convergé, la solution stable de (IV.74) est: Si $\lambda < 1 \Rightarrow E\{\Delta \mathbf{w}(\infty)\} = \mathbf{0}_L$. D'où le vecteur moyen des coefficients du filtre adaptatif est :

$$E\{\mathbf{w}(\infty)\} = \mathbf{w}_{\text{opt}L} \quad (\text{IV.75})$$

IV.5 Résultats de simulations

Pour confirmer la validité de notre analyse et montrer les performances des algorithmes proposés, quelques simulations sont effectuées. Le facteur d'oubli λ et le facteur de fuite η de l'algorithme M-SMFTF sont choisis selon (IV.65) avec un signal d'entrée stationnaire. La constante de régulation c_a est utilisée pour limiter la dynamique du gain d'adaptation.

IV.5.1 Comparaison des performances des algorithmes SMFTF et M-SMFTF

Nous définissons la norme de la différence des gains par $Ndg(n)$ [89, 91]. Cette variable est employée dans nos simulations pour vérifier l'égalité des expressions des variables de vraisemblances. Elle est calculée par:

$$Ndg(n) = 10 \log_{10} \left(E \left\{ \left\| \Delta \mathbf{g}_{L,n} \right\|^2 \right\} \right) \quad (\text{IV.76})$$

où $\Delta \mathbf{g}_{L,n} = (\gamma_{L,n}^d \tilde{\mathbf{k}}_{L,n} - \gamma_{L,n}^f \tilde{\mathbf{k}}_{L,n})$ est un vecteur représentant la différence des gains, $\gamma_{L,n}^d$ et $\gamma_{L,n}^f$ sont les variables de vraisemblances calculées dans les algorithmes SMFTF (utilisant directement la définition) et M-SMFTF (utilisant la formule récursive) respectivement. Nous avons simulé les algorithmes pour vérifier leur exactitude. Le signal d'entrée x_n utilisé dans notre simulation est un bruit blanc gaussien, de moyenne nulle et de variance unité. Pour un ordre de filtre L donné, nous exécutons les algorithmes SMFTF et M-SMFTF avec les mêmes paramètres, à savoir, un facteur d'oubli λ ($\lambda \geq (1 - 1/L)$), un facteur de fuite η et une constante de régulation c_a . La Fig. IV.2 décrit les évolutions en décibels des normes de la différence des gains $Ndg(n)$ pour les ordres $L=32$ et $L=256$. Nous pouvons voir que le signal approximatif d'erreur reste constant et très faible. Les algorithmes M-SMFTF (6L) et SMFTF (7L) produisent exactement le même signal d'erreur de filtrage.

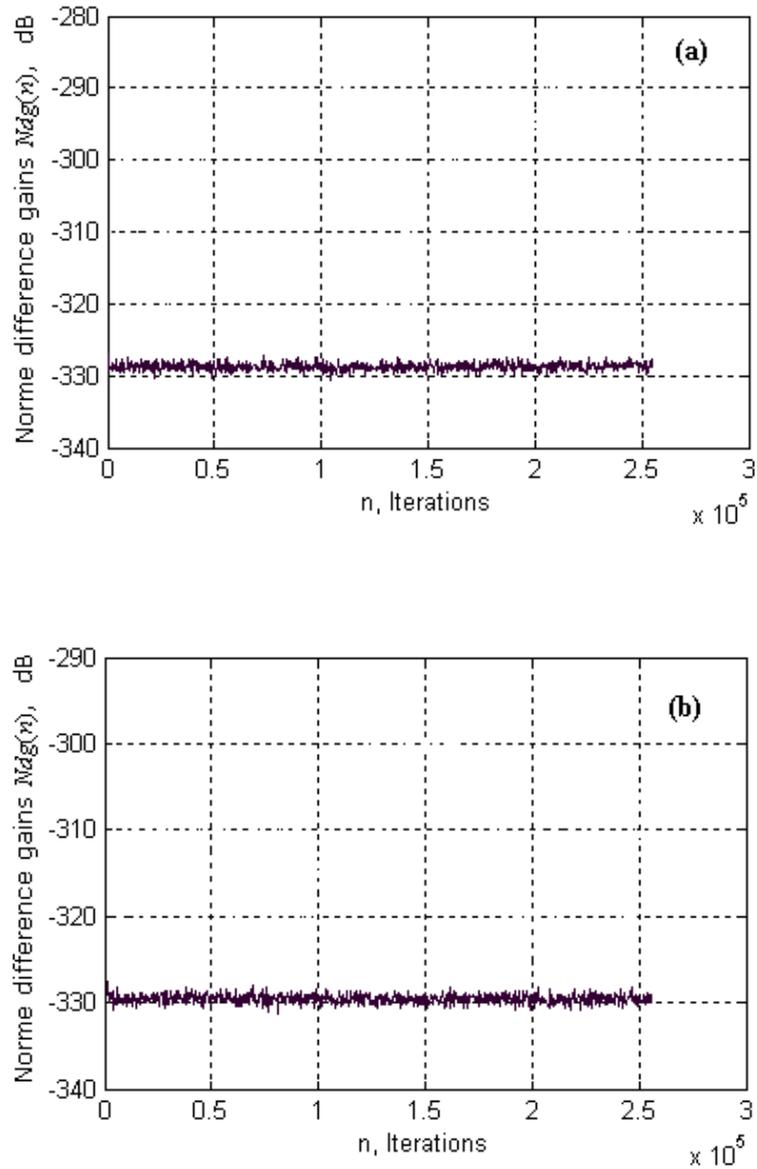


Fig. IV.2: Evolution de la norme de la différence des gains $Ndg(n)$
(a): $L=32$, $\lambda=0.9688$, $\eta=0.98$, $c_a=0.1$, $E_0=0.5$;
(b): $L=256$, $\lambda=0.9961$, $\eta=0.995$, $c_a=0.1$, $E_0=3$.

IV.5.2 Comportement des algorithmes M-SMFTF et RM-SMFTF

Le signal d'entrée x_n utilisé dans notre simulation est un bruit blanc gaussien, de moyenne nulle et de variance unité. La performance d'évaluation est mesurée par l'erreur quadratique moyenne $MSE(n)$.

IV.5.2.1 Influence des valeurs η et c_a sur l'algorithme M-SMFTF

Pour un filtre de taille L , le facteur d'oubli est choisi comme la borne inférieure de la stabilité numérique de l'algorithme M-SMFTF (IV.65), soit $\lambda = (1 - 1/L)$.

La Fig. IV.3 montre l'effet du paramètre η sur le comportement de l'algorithme M-SMFTF pour les tailles $L=32$ et $L=256$. On remarque que plus la valeur de η est grande, plus l'algorithme est meilleur du point de vue vitesse de convergence.

La Fig. IV.4 montre l'effet du paramètre c_a sur le comportement de l'algorithme M-SMFTF pour les tailles $L=32$ et $L=256$. On remarque que plus la valeur de c_a est petite, plus l'algorithme est meilleur du point de vue vitesse de convergence mais l'erreur de filtrage est plus grande.

IV.5.2.2 Influence de la valeur η sur l'algorithme RM-SMFTF

Pour un filtre de taille L et un prédicteur d'ordre P , le facteur d'oubli est choisi comme la borne inférieure de la stabilité numérique de l'algorithme RM-SMFTF, soit $\lambda = (1 - 1/P)$.

La Fig. IV.5 montre l'effet du paramètre η sur le comportement de l'algorithme RM-SMFTF pour un filtre de taille $L=256$ et les ordres respectifs $P=16$ et $P=32$. On remarque que plus la valeur de η est grande, plus l'algorithme est meilleur du point de vue vitesse de convergence.

Les mêmes remarques que l'algorithme M-SMFTF pour la valeur de c_a .

En résumé, la valeur de η est plus liée à la vitesse de convergence que l'erreur de filtrage, tandis que la valeur de c_a est liée à la vitesse de convergence et l'erreur de filtrage. En plus, une valeur de η trop proche de 1 et/ou une valeur de c_a trop petite risque de rendre l'algorithme instable. Par conséquent, il faut bien choisir ces valeurs pour que l'algorithme soit stable, afin d'avoir une meilleure vitesse de convergence et une erreur de filtrage la plus petite possible.

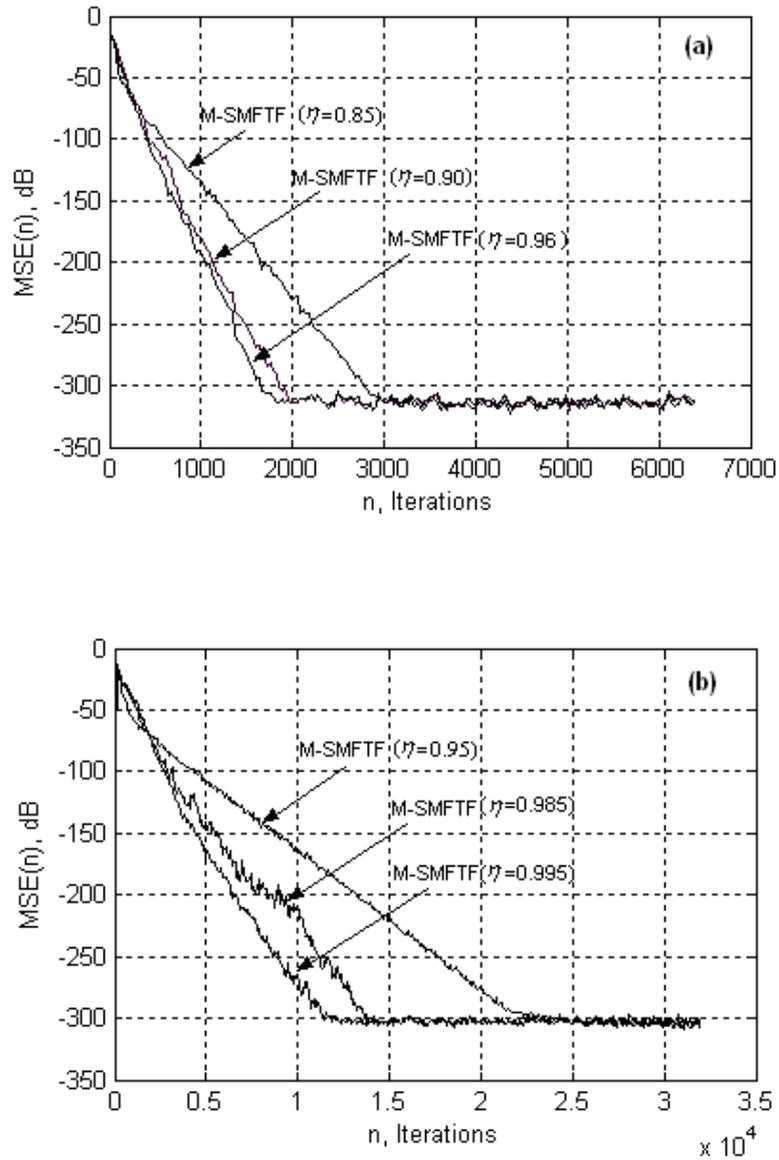


Fig. IV.3 : Influence de la valeur η sur l'algorithme M-SMFTF

(a) : $L=32$, $\lambda=0.9688$, $c_a=0.1$, $E_0=0.5$;

(b) : $L=256$, $\lambda=0.9961$, $c_a=0.1$, $E_0=3$.

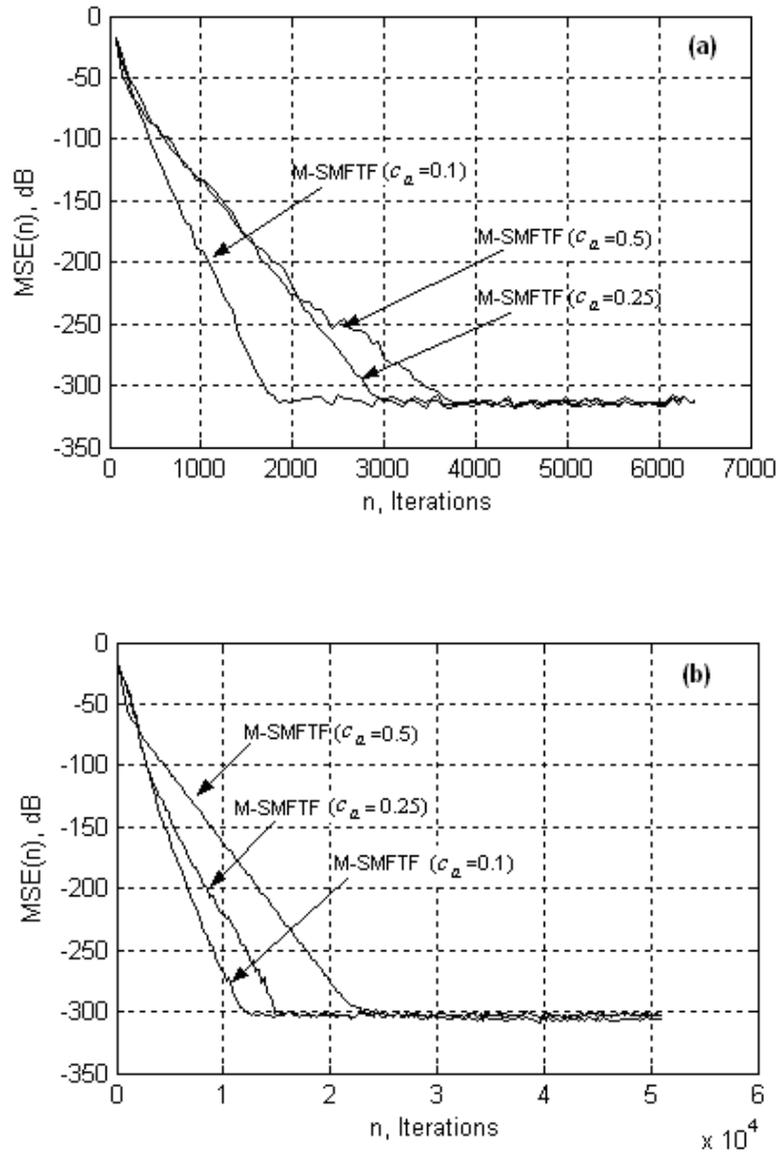


Fig. IV.4 : Influence de la valeur c_a sur l'algorithme M-SMFTF
 (a) : $L=32$, $\lambda = 0.9688$, $\eta = 0.96$, $E_0=0.5$;
 (b) : $L=256$, $\lambda = 0.9961$, $\eta = 0.995$, $E_0=3$.

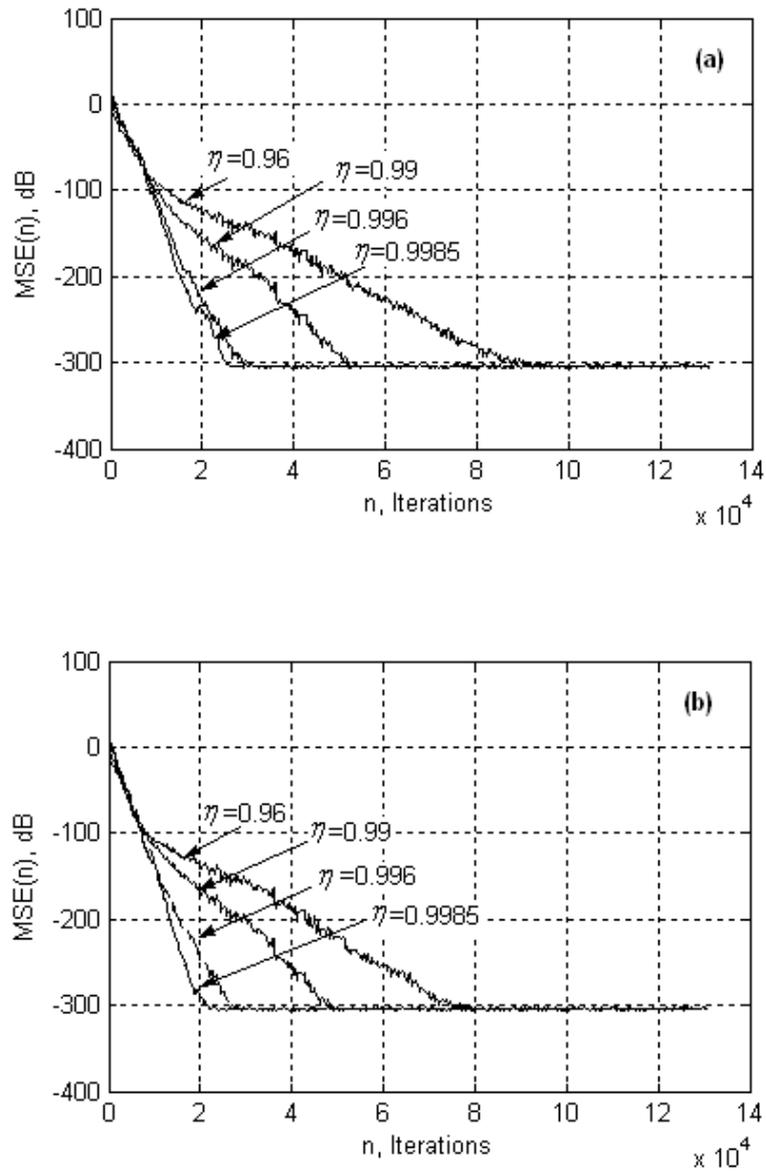


Fig. IV.5: Influence de la valeur η sur l'algorithme RM-SMFTF pour $L=256$

(a) : $P=16$, $\lambda=0.9375$, $c_a=0.1$, $E_0=0.25$;

(b) : $P=32$, $\lambda=0.9688$, $c_a=0.1$, $E_0=0.5$.

IV.5.3 Comparaison des performances du M-SMFTF avec le NS-FRLS et le NLMS

Le signal d'entrée x_n est un signal stationnaire très corrélé (USASI), avec ou sans bruit additif sur la sortie. Dans cette simulation, nous comparons les performances de l'algorithme M-SMFTF avec les algorithmes NLMS et NS-FRLS. Le pas d'adaptation de l'algorithme NLMS est choisi $\mu=1$, afin d'obtenir la meilleure vitesse de convergence possible. Pour l'algorithme NS-FRLS, le facteur d'oubli est choisi comme la borne inférieure de stabilité numérique afin d'obtenir la meilleure capacité de poursuite, soit $\lambda = (1 - 1/3L)$. Nous représentons les résultats sous formes de courbes.

Les Fig. IV.6 et Fig. IV.7 présentent les résultats obtenus pour des filtres de taille $L=32$ et $L=256$ respectivement. Pour le cas sans bruit additif (Fig. IV.6a et Fig. IV.7a), nous remarquons que, l'algorithme M-SMFTF converge beaucoup plus rapidement que l'algorithme NLMS.

Dans les Fig. IV.6b et Fig. IV.7b, nous avons ajouté un bruit blanc en sortie avec un rapport signal à bruit $SNR=50\text{dB}$. Ces résultats prouvent que l'algorithme proposé réalise de meilleures performances que l'algorithme NLMS, et ne présente aucune dégradation apparente de performance comparée à l'algorithme NS-FRLS.

Il y a seulement une variation dans l'erreur quadratique moyenne finale $MSE(n)$ pour les trois algorithmes, en raison du bruit en sortie $SNR=50\text{dB}$ (Fig. IV.6b et Fig. IV.7b). Les différences dans le $MSE(n)$ finale pour les algorithmes M-SMFTF et NS-FRLS sont dues à l'utilisation de différents facteurs oubliés λ .

Nous avons simulé un changement brusque de la réponse impulsionnelle en multipliant le signal désiré par 1.5 à partir d'un certain nombre d'échantillons. La Fig.IV.8 (resp. Fig.IV.9) compare les performances adaptatives du point de vue de vitesse de convergence pour un filtre de taille $L=32$ (resp. $L=256$).

Les graphes (Fig.IV.8, Fig.IV.9) prouvent que de meilleures performances dans la vitesse de convergence sont obtenues avec l'algorithme M-SMFTF.

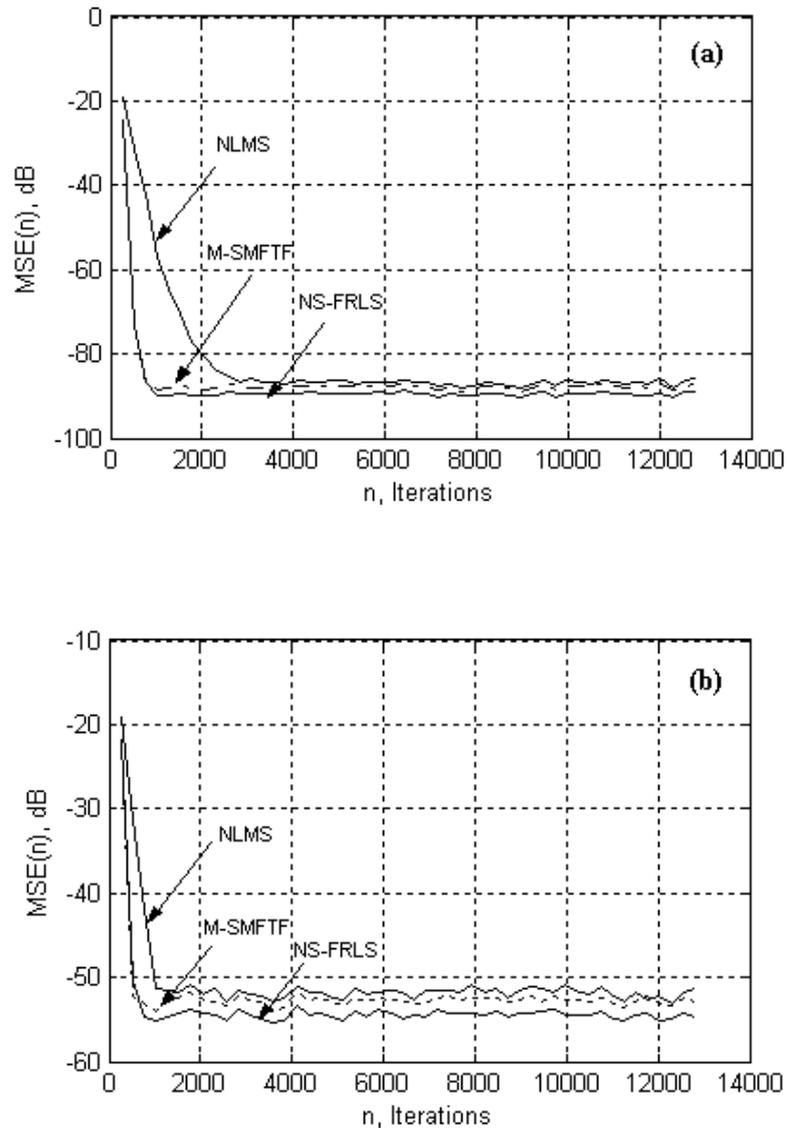


Fig. IV.6 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS pour un signal d'entrée USASI et $L=32$,

(a) : sans bruit en sortie ; (b) : avec bruit en sortie $SNR=50\text{dB}$

M-SMFTF: $\lambda=0.9688$, $\eta=0.96$, $c_a=0.1$, $E_0=0.5$; NS-FRLS: $\lambda=0.9896$; NLMS: $\mu=1$.

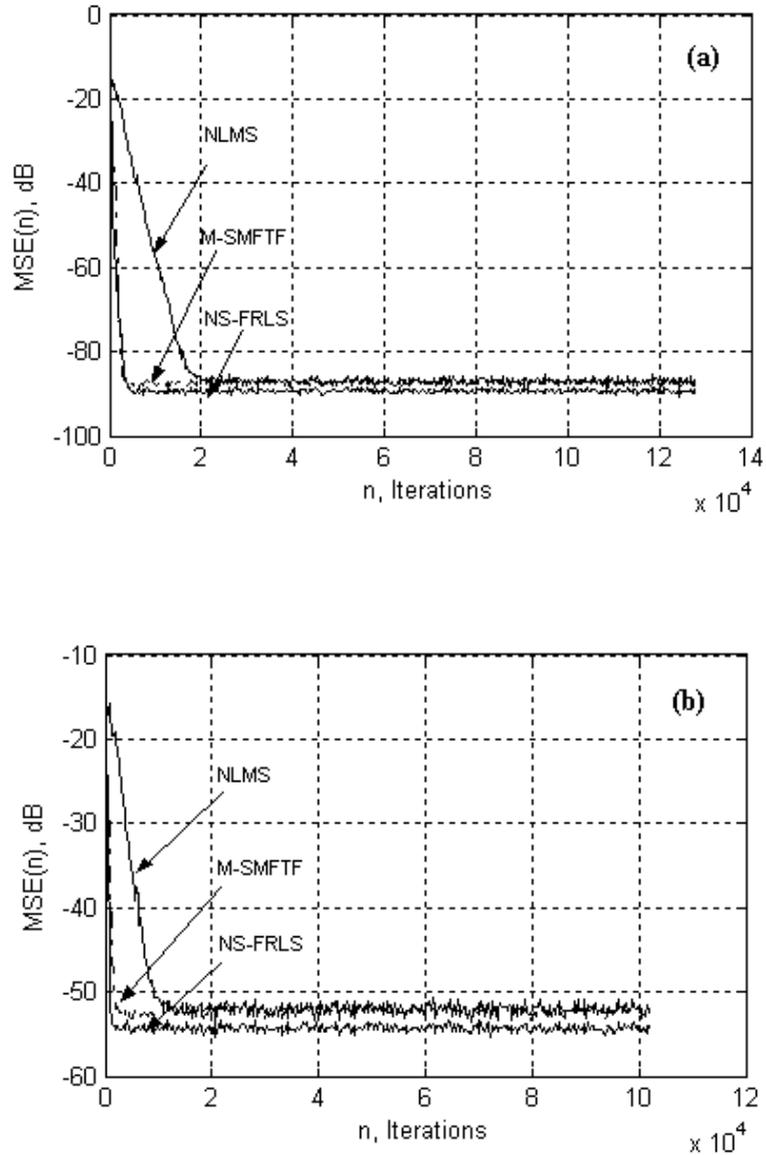


Fig. IV.7 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS pour un signal d'entrée USASI et $L=256$,
 (a) : sans bruit en sortie ; (b) : avec bruit en sortie $SNR=50$ dB
 M-SMFTF: $\lambda = 0.9961$, $\eta = 0.995$, $c_a = 0.1$, $E_0 = 1$; NS-FRLS: $\lambda = 0.9987$; NLMS: $\mu = 1$.

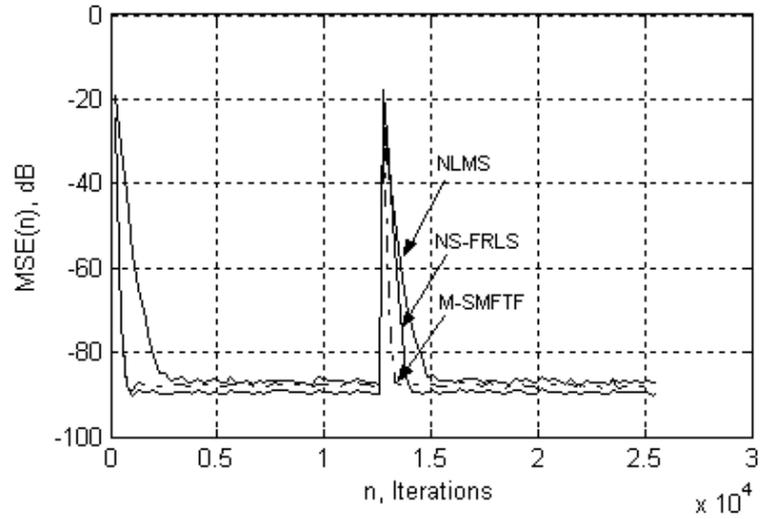


Fig. IV.8 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS pour un signal d'entrée USASI et $L=32$ avec une non stationnarité du système, M-SMFTF: $\lambda=0.9688$, $\eta=0.96$, $c_a=0.1$, $E_0=0.5$; NS-FRLS: $\lambda=0.9896$; NLMS: $\mu=1$

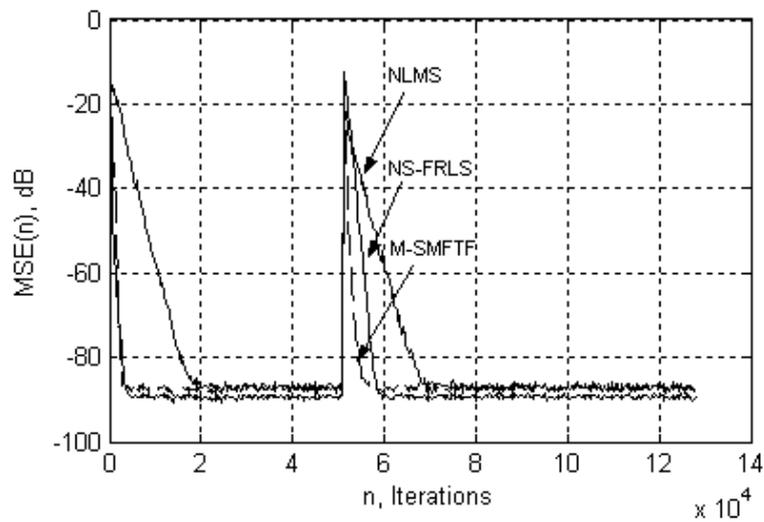


Fig. IV.9 : Comparaison des performances de M-SMFTF avec NS-FRLS et NLMS pour un signal d'entrée USASI et $L=256$ avec une non stationnarité du système, M-SMFTF: $\lambda=0.9961$, $\eta=0.995$, $c_a=0.1$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$

IV.5.4 Effet de l'ordre du prédicteur P sur l'algorithme RM-SMFTF

Nous simulons un changement brusque de la réponse impulsionnelle en multipliant le signal désiré par 1.5 à partir d'un certain nombre d'échantillons. La Fig. IV.10 montre l'effet de l'ordre du prédicteur P sur le comportement de l'algorithme RM-SMFTF pour un signal d'entrée stationnaire et un filtre de taille $L=256$. Si l'ordre du prédicteur P est égal à la taille du filtre L , on retrouve l'algorithme M-SMFTF. Pour un ordre du prédicteur P , la condition (IV.65) devient $\lambda \geq (1 - 1/P)$. La simulation montre que la vitesse de convergence augmente avec l'ordre du prédicteur P , c'est-à-dire lorsque l'ordre du prédicteur P diminue, les performances de l'algorithme RM-SMFTF deviennent moins bonne et ce, à cause de l'extrapolation de la matrice de covariance qui devient mauvaise. On peut dire que l'algorithme donne de très bonnes performances adaptatives pour une extrapolation d'ordre raisonnable. L'algorithme RM-SMFTF reste toujours meilleur que l'algorithme NLMS.

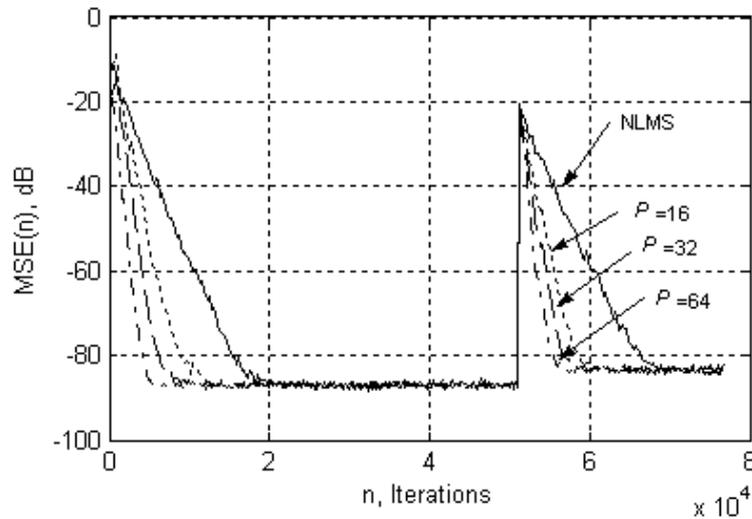


Fig. IV.10: Effet de l'ordre du prédicteur P sur l'algorithme RM-SMFTF pour un signal d'entrée USASI et $L=256$ avec une non stationnarité du système,

$$P=16, \lambda=0.9375, \eta=0.99, c_a=0.1, E_0=0.1;$$

$$P=32, \lambda=0.9688, \eta=0.99, c_a=0.1, E_0=0.25;$$

$$P=64, \lambda=0.9844, \eta=0.99, c_a=0.1, E_0=0.5.$$

$$\text{NLMS: } \mu=1$$

IV.6 Conclusion

Nous avons proposé un nouvel algorithme rapide de type FTF simplifié et modifié (M-SMFTF) pour le filtrage adaptatif. Cet algorithme possède une complexité de calcul égale à $6L$, et celle-ci est réduite à $(2L+4P)$ par l'algorithme RM-SMFTF en utilisant un prédicteur aller réduit d'ordre P , où P est une valeur très petite devant la taille du filtre L ($P \ll L$). Ceci est très intéressant pour l'adaptation de filtres très longs comme ceux utilisés dans les problèmes de l'annulation d'écho acoustique.

Nous avons décrit les différentes étapes pour l'obtention de l'algorithme M-SMFTF. Nous avons proposé la réduction de l'ordre du prédicteur aller par une méthode d'extrapolation qui consistait à construire la matrice d'autocorrélation d'ordre L à partir d'une estimation de la matrice d'autocorrélation d'ordre P . Nous avons étudié la propagation des erreurs de toutes les quantités récursives de la partie prédiction de l'algorithme proposé. Nous avons ensuite analysé les propriétés numériques de l'algorithme. Cet algorithme est stable numériquement pour un choix convenable du facteur de fuite et une condition sur le facteur d'oubli. Ces analyses ont montré la supériorité en termes de capacité de poursuite de cet algorithme par rapport à l'algorithme NS-FRLS. Nous avons comparé, par simulations, les performances adaptatives de l'algorithme proposé à celles du NS-FRLS et du NLMS dans plusieurs situations. Les simulations ont prouvé la supériorité de l'algorithme proposé.

Chapitre V

APPLICATION A L'ANNULATION D'ECHO ACOUSTIQUE

V.1 Introduction

Le principal objectif de notre étude, est d'évaluer la possibilité de développer des algorithmes d'adaptation pour les applications de système d'annulation d'écho acoustique. Notre étude a donc été menée pour améliorer les performances de ce système et de diminuer sa complexité de calcul au maximum, tout en maintenant un équilibre entre son coût d'implantation et ses performances adaptatives.

Nous avons proposé, au chapitre précédent (Chap.IV), l'algorithme M-SMFTF pour le filtrage adaptatif. Cet algorithme est stable numériquement et très performant quand le signal d'entrée est stationnaire avec bruit additif sur le signal de sortie. Dans ce chapitre, nous comparons les performances adaptatives de l'algorithme M-SMFTF avec les algorithmes NS-FRLS (version numériquement stable du FRLS, Chap.III) et NLMS [92]. La comparaison sera faite dans plusieurs situations. Les simulations sont effectuées lorsque le signal d'excitation est stationnaire, très corrélé et où le système à identifier est non stationnaire. Nous évaluons par la suite, les performances adaptatives de l'algorithme dans des situations réelles où le signal d'entrée est de la parole avec un bruit additif en sortie. Nous décrivons les différents signaux utilisés et les critères de performances mesurés par l'erreur quadratique moyenne (*MSE*) et le rapport de la puissance de l'écho à celle de l'écho résiduel (*ERLE*).

V.2 Principe de l'annulation d'écho acoustique

Il existe un fort couplage acoustique entre le haut-parleur et le microphone à l'origine de la transmission de l'écho vers le locuteur distant. Cet écho constitue une part très importante de la gêne occasionnée pour ce dernier dans une communication de type mains-libres. Il importe donc, dans l'élaboration d'un système mains-libres, de prendre en compte ce couplage et de proposer des solutions à l'annulation de l'écho présent dans le signal transmis [93-95].

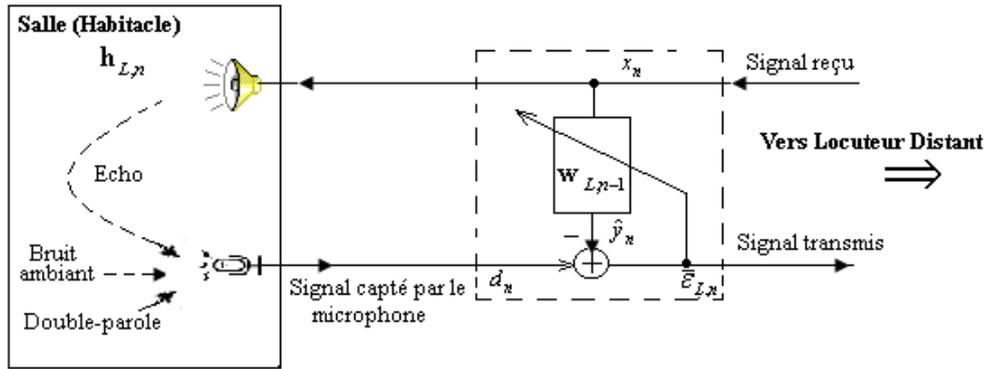


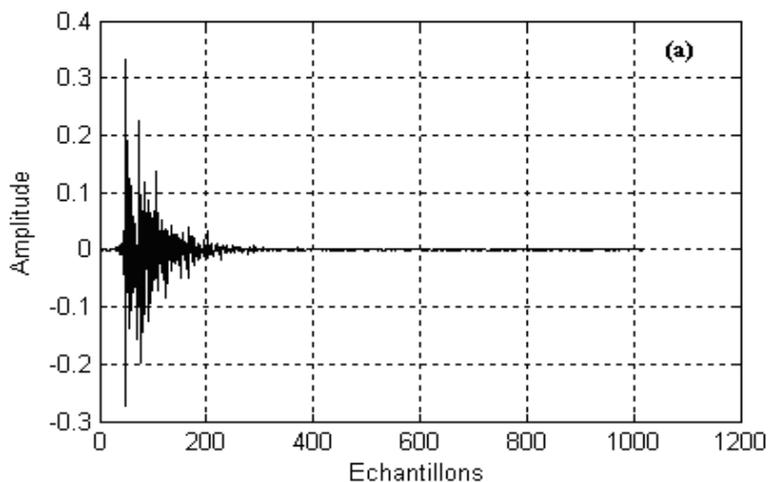
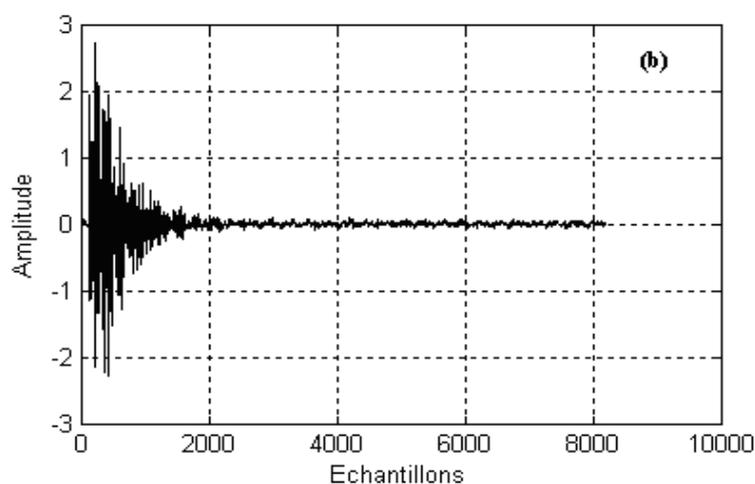
Fig. V.1: Système d'annulation d'écho

La Fig.V.1 décrit le traitement d'annulation d'écho acoustique par filtrage adaptatif [92, 93]. Le signal x_n est diffusé par le haut-parleur dans l'espace acoustique (salle, habitacle, ...). Ce signal est réfléchi par les parois et les différents objets pour former le signal d'écho y_n capté par le microphone. Le signal y_n est modélisé par un filtrage linéaire du signal diffusé x_n . La réponse impulsionnelle $\mathbf{h}_{\infty,n}$, caractérisant le chemin acoustique réel entre le haut-parleur et le microphone, est causale et de longueur infinie. Le signal capté par le microphone d_n est alors la somme de l'écho y_n , du bruit ambiant v_n de l'espace acoustique (bruit de salle, bruit du véhicule, ...), ainsi que d'un éventuel signal utile s_n (généralement le signal de parole à transmettre, appelé signal double parole).

$$d_n = y_n + s_n + v_n \quad (\text{V.1})$$

Le rôle de l'annuleur d'écho est d'estimer le signal d'écho y_n , en identifiant le chemin acoustique. Cette estimation est ensuite soustraite du signal d_n pour délivrer un signal exempt d'écho. L'énergie de la réponse impulsionnelle réelle $\mathbf{h}_{\infty,n}$ décroît de façon exponentielle et les coefficients d'indice élevé sont négligeables. Dans la pratique, on estime la réponse tronquée $\mathbf{h}_{L,n}$ à L coefficients qui contiennent la majeure partie de l'énergie de la réponse réelle infinie.

On supposera que le canal à identifier peut être modélisé par un filtre linéaire $\mathbf{h}_{L,n}$ dont les coefficients caractérisent les différents couples retard/atténuation. La dimension de ce filtre dépend de l'environnement du mains-libres et de la fréquence d'échantillonnage F_e (Fig.V.2).


 Fig. V.2a : Réponse impulsionnelle d'un habitacle de voiture ($F_e=16\text{kHz}$)

 Fig. V.2b : Réponse impulsionnelle d'une salle de téléconférence ($F_e=16\text{kHz}$)

Le principe du filtrage adaptatif est donc d'identifier, à chaque instant n , la réponse impulsionnelle tronquée $\mathbf{h}_{L,n}$ à l'aide d'un filtre linéaire RIF $\mathbf{w}_{L,n}$ de taille L . Le signal x_n est filtré par $\mathbf{w}_{L,n}$ pour fournir une estimation \hat{y}_n du signal d'écho. L'estimation est soustraite du signal d_n capté par le microphone pour former le signal d'erreur $\bar{\varepsilon}_{L,n}$:

$$\bar{\varepsilon}_{L,n} = d_n - \hat{y}_n \quad (\text{V.2})$$

On différencie deux cas, selon qu'un signal utile s_n est présent ou non sur le signal du microphone d_n :

- S'il existe un signal utile, on a :

$$\bar{\varepsilon}_{L,n} = (y_n - \hat{y}_n) + s_n + v_n \quad (\text{V.3})$$

On parle alors de phase de double parole. Le signal d'erreur $\bar{\varepsilon}_{L,n}$ intègre un signal qui perturbe la mise à jour du filtre. Généralement, les coefficients du filtre $\mathbf{w}_{L,n}$ ne sont pas corrigés (adaptation figée) durant cette phase, ou bien, cette correction tient compte de cette perturbation (adaptation ralentie ou contrôlée) [96].

- S'il n'existe pas de signal utile, on a :

$$\bar{\varepsilon}_{L,n} = (y_n - \hat{y}_n) + v_n \quad (\text{V.4})$$

Le signal d'erreur $\bar{\varepsilon}_{L,n}$ n'est composé que de l'erreur d'estimation du signal d'écho bruité par le signal v_n . Si le bruit est négligeable, le signal résiduel $\bar{\varepsilon}_{L,n}$ est utilisé pour corriger le filtre $\mathbf{w}_{L,n}$ afin de réduire l'erreur d'estimation future. Si v_n constitue une perturbation trop importante, des techniques de débruitage peuvent être mises en œuvre pour fournir un signal d'erreur débruité qui est alors utilisé pour la correction du filtre [97].

Notons que l'on ne s'intéresse ici qu'à l'aspect annulation d'écho.

V.3 Descriptions des critères et des signaux utilisés

L'annulation d'écho est réalisée en identifiant la fonction de transfert $\mathbf{h}_{L,n} = \mathbf{w}_{\text{opt}L}$ de l'ensemble par un filtre adaptatif [95] représenté sur la Fig.IV.1 par $\mathbf{w}_{L,n-1}$. L'écho est le résultat de la convolution entre ce filtre et le signal d'excitation du haut-parleur :

$$y_n = \mathbf{w}_{\text{opt}L}^T \mathbf{x}_{L,n} \quad (\text{V.5})$$

Notons que la réponse impulsionnelle du canal n'est pas obligatoirement fixe mais peut varier sensiblement au cours d'une communication. Pour de raisons de lisibilité, l'index temporel n pour $\mathbf{w}_{\text{opt}L}$ n'est pas reproduit. Le canal est estimé par le filtre adaptatif dont la dimension dépend de la nature de l'application (Fig.V.2). L'écho estimé est donné par la convolution :

$$\hat{y}_n = \mathbf{w}_{L,n-1}^T \mathbf{x}_{L,n} \quad (\text{V.6})$$

L'objectif de l'algorithme adaptatif d'annulation d'écho est de fournir à chaque instant n une estimée la plus fidèle possible du canal acoustique. Cette estimation est réalisée en minimisant une fonction de coût qui dépend des coefficients du filtre adaptatif et plus généralement de l'erreur en sortie du filtre $\bar{\varepsilon}_{L,n}$. L'erreur quadratique moyenne $MSE(n)$ est définie par :

$$MSE(n) = E\{\bar{\varepsilon}_{L,n}^2\} \quad (\text{V.7})$$

On caractérise la réduction d'écho par la mesure *ERLE* (Echo Return Loss Enhancement) qui représente l'amélioration des pertes de retour d'écho. Elle est définie comme le rapport de la puissance de l'écho réel à celle de l'écho résiduel :

$$ERLE(n) = \frac{E\{y_n^2\}}{E\{(y_n - \hat{y}_n)^2\}} \quad (V.8)$$

Dans la plupart des situations, le signal d'écho et l'écho résiduel ne sont pas disponible, aussi les remplace-t-on respectivement par le signal capté par le microphone et le signal transmis. On utilisera alors dans ces cas l'expression de l' *ERLE* suivante :

$$ERLE(n) = \frac{E\{d_n^2\}}{E\{\mathcal{E}_{L,n}^2\}} \quad (V.9)$$

Nous utilisons trois différents types de signaux d'entrée, qui sont ensuite filtrés par deux différentes réponses impulsionnelles du canal à identifier, données par les Fig.V.2a et Fig.V.2b tronquées à 256 et 1500 points respectivement, pour obtenir les signaux désirés. La fréquence d'échantillonnage des signaux est de 16kHz.

Le premier signal d'entrée est un bruit corrélé stationnaire avec un spectre équivalent au spectre moyen de la parole, appelé bruit USASI dans le domaine de l'annulation d'écho acoustique. Ce signal, de moyenne nulle et de variance égale à 0.32. . Sa dynamique spectrale est de 29 dB.

Le second est un signal stationnaire, fortement corrélé, généré par un modèle AR d'ordre 20 déduit de l'analyse LPC de la voyelle « U » [10]. Son spectre de fréquence est représenté sur le schéma de la Fig.V.3 où nous pouvons voir quatre crêtes principales. Sa dynamique spectrale est environ de 40dB.

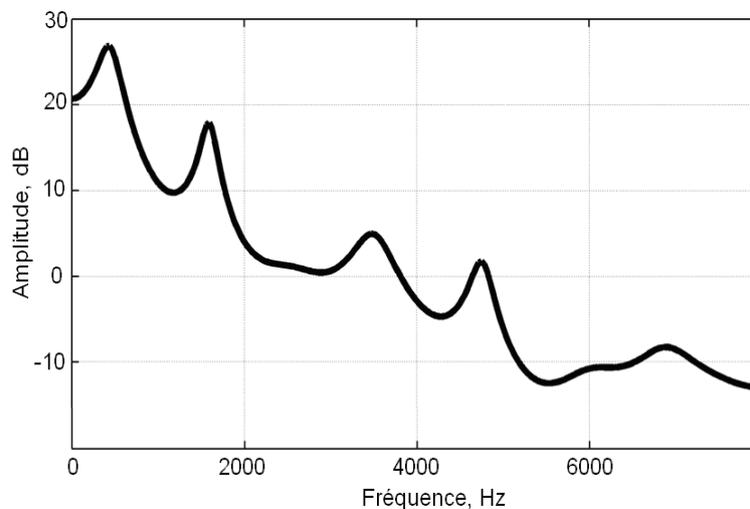


Fig.V.3: Spectre du modèle AR(20)

Le troisième signal d'entrée est un signal de parole réel (Fig.V.4), signal non stationnaire, obtenue par concaténation d'une voix féminine et d'une voix masculine dans la même séquence. Sa dynamique spectrale est estimée à 40dB.

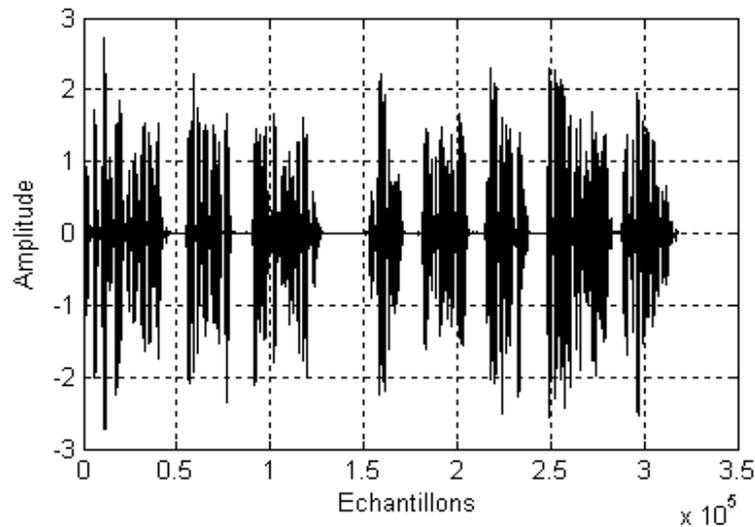


Fig.V.4 : Signal de parole ($F_e=16\text{kHz}$)

V.4 Comparaison des performances des différents algorithmes

Pour l'estimation en temps réel, il existe deux critères concernant le choix d'un algorithme adaptatif. Le premier concerne les performances intrinsèques de l'algorithme, c'est-à-dire sa vitesse de convergence vers le filtre optimal de Wiener, sa capacité de poursuivre et sa robustesse au bruit additif sur la sortie. Le second critère concerne les contraintes externes de l'application, et englobe la complexité de calcul, la stabilité numérique.

Le tableau V.1, résume la complexité de calcul et les conditions de stabilité des algorithmes NLMS, NS-FRLS, M-SMFTF et RM-SMFTF, étudiés aux chapitres précédents pour le cas d'un bruit blanc en entrée. Si nous comparons que les deux algorithmes (M-SMFTF et NS-FRLS), on voit que la complexité de calcul pour l'algorithme M-SMFTF est de $6L$ alors que celle de l'algorithme NS-FRLS est de $8L$. Pour la stabilité numérique, l'algorithme M-SMFTF est stable sous la condition suivante $\lambda \geq (1-1/L)$ pour un signal d'entrée quelconque (stationnaire ou non stationnaire) et sur des durées longues. Alors que pour l'algorithme NS-FRLS, la condition est $\lambda \geq (1-1/3L)$ et cela pour un signal d'entrée stationnaire. Mais pour un signal d'entrée non stationnaire comme la parole, des instabilités numériques ont été observées. A noter que pour un

facteur d'oubli très proche de 1, par exemple $\lambda \geq (1-1/10L)$ pour L assez grand ($200 \leq L \leq 2000$), nous n'avons pas de problème d'instabilité sur le signal de parole pour des durées pas trop longue, mais pour une durée assez longue une tendance à l'instabilité est observé. Par conséquent, l'algorithme NS-FRLS sera pénalisé en vitesse de convergence et en capacité de poursuite.

Pour évaluer les performances adaptatives de ses algorithmes, nous nous plaçons dans un contexte d'identification adaptative correspondant à l'annulation d'écho acoustique. Plusieurs simulations numériques ont été réalisées pour évaluer et comparer les performances des différents algorithmes du filtrage adaptatif (NLMS, NS-FRLS, M-SMFTF). Pour ces tests, l'algorithme M-SMFTF est comparé aux autres algorithmes, NLMS et NS-FRLS.

NLMS	NS-FRLS	M-SMFTF	RM-SMFTF
$2L$	$8L$	$6L$	$2L+4P$
$0 < \mu < 2$	$\lambda \geq (1-1/3L)$	$\lambda \geq (1-1/L)$	$\lambda \geq (1-1/P)$

Tableau V.1 : Complexité de calcul et Condition de stabilité des algorithmes étudiés
 L : Taille du filtre, P : Ordre du prédicteur aller ($P \ll L$)

V.5 Résultats de simulations

V.5.1 Canal stationnaire avec entrée stationnaire

Le signal désiré est obtenu à partir du signal USASI filtré par la réponse impulsionnelle de voiture tronquée à 256 points et en additionnant un bruit blanc gaussien de rapport signal à bruit SNR . La même longueur ($L=256$) est utilisée pour le filtre adaptatif. La performance d'évaluation est mesurée par l'erreur quadratique moyenne $MSE(n)$ [98].

Dans la Fig.V.5, nous comparons les performances de l'algorithme M-SMFTF à celles des algorithmes NS-FRLS et NLMS. Pour l'algorithme M-SMFTF, le facteur d'oubli est $\lambda = 0.9975$, le facteur de fuite est $\eta = 0.995$ et la constante de régularisation est choisi comme $c_a = 0.1$. Le facteur d'oubli de l'algorithme NS-FRLS est $\lambda = 0.9987$. La simulation de la Fig.V.6 montre les performances adaptatives de l'algorithme M-SMFTF dans la même situation que celle de la Fig.V.5 quand il y a un bruit blanc additif sur la sortie avec $SNR=50dB$. Ces résultats prouvent

que l'algorithme M-SMFTF réalise de meilleures performances que l'algorithme NLMS, et ne présente aucune dégradation apparente de performance comparée à l'algorithme NS-FRLS.

La Fig.V.7 donne les performances des algorithmes RM-SMFTF et NLMS. L'ordre du prédicteur aller de l'algorithme RM-SMFTF est $P=32$, son facteur d'oubli est $\lambda=0.9688$, les paramètres sont $\eta=0.9985$ et $c_a=0.5$. La même simulation mais avec un bruit blanc additif sur la sortie avec un $SNR=50$ dB est reproduite sur la Fig.V.8. Nous remarquons que l'algorithme RM-SMFTF adapte mieux que l'algorithme NLMS.

V.5.2 Canal non stationnaire avec entrée stationnaire

Pour comparer expérimentalement les algorithmes NLMS, NS-FRLS avec l'algorithme proposé M-SMFTF du point de vue de la vitesse de convergence et de la capacité de poursuite, nous considérons le cas où le signal d'entrée est stationnaire, très corrélé et où le système à identifier est non stationnaire [98]. Le pas d'adaptation du NLMS est choisi $\mu=1$, afin d'obtenir la meilleure vitesse de convergence possible. Pour l'algorithme NS-FRLS, le facteur d'oubli est choisi comme la borne inférieure de sa stabilité numérique afin d'obtenir la meilleure capacité de poursuite possible, soit $\lambda=1-1/3L$ où L est la taille du filtre. Pour l'algorithme M-SMFTF, le facteur d'oubli est choisi comme suit $\lambda=1-1/L$ et enfin pour l'algorithme RM-SMFTF, le facteur d'oubli est choisi égale à $\lambda=1-1/P$ où P est l'ordre du prédicteur aller. Afin de tester la capacité de poursuite de l'algorithme en plus de la vitesse de convergence, nous introduisons la non stationnarité du système par modification linéaire du gain en sortie. Les résultats des simulations de cette comparaison sont mesurés par les critères de performances $MSE(n)$ et $ERLE(n)$.

Les résultats des Fig.V.9 à Fig.V.12, sont obtenus dans le cas d'un bruit USASI en entrée et un filtre de taille $L=256$ où la réponse impulsionnelle du système à identifier est les 256 premiers coefficients de la réponse impulsionnelle de l'habitacle de voiture. Les Fig.V.13 à Fig.V.16, sont obtenues pour les mêmes conditions de simulations mais avec un signal d'entrée généré par un modèle AR(20). On voit que les algorithmes M-SMFTF et RM-SMFTF convergent plus vite et poursuivent mieux la variation du système à identifier que l'algorithme NLMS avec la même erreur résiduelle. On peut voir aussi, que la convergence initiale est presque la même pour les algorithmes M-SMFTF et NS-FRLS. Mais l'algorithme M-SMFTF réalise une meilleure re-convergence après le changement brusque de la réponse impulsionnelle. On remarque aussi que l'erreur résiduelle de l'algorithme M-SMFTF est légèrement supérieure à celle du NS-FRLS

Les Fig.V.17 à Fig.V.20 montrent les performances adaptatives des algorithmes M-SMFTF et RM-SMFTF à celles des algorithmes NS-FRLS et NLMS, quand le signal d'entrée est un bruit USASI et le système à identifier est les 1500 premiers coefficients de la réponse impulsionnelle d'une salle de téléconférence. La taille du filtre adaptatif est $L=1500$. On voit que l'algorithme M-SMFTF poursuit mieux que l'algorithme NS-FRLS et on observe aussi que l'algorithme RM-SMFTF réalise de meilleures performances que le NLMS, tant du point de vue de la vitesse de convergence que de l'erreur résiduelle.

A partir de ces simulations, nous constatons que l'algorithme M-SMFTF fournit une meilleure convergence par rapport aux autres algorithmes, NLMS et NS-FRLS. Il est alors intéressant d'utiliser cet algorithme pour rendre au maximum l'écho résiduel inaudible à la sortie du système d'annulation d'écho acoustique.

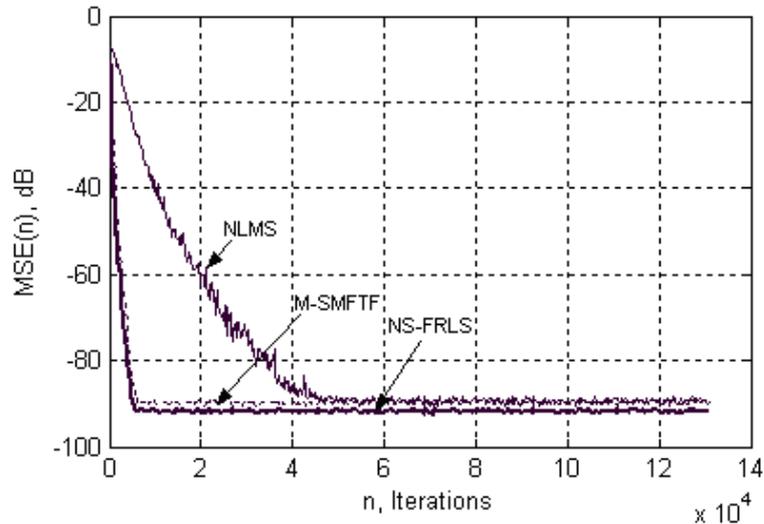


Fig.V.5: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie
M-SMFTF: $\lambda=0.9975$, $\eta=0.995$, $c_a=0.1$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

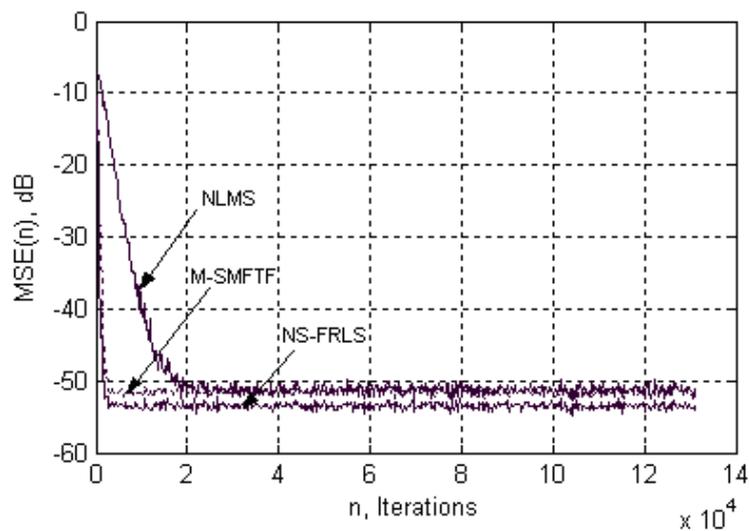


Fig.V.6: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, avec bruit additif sur la sortie ($SNR=50dB$).
M-SMFTF: $\lambda=0.9975$, $\eta=0.995$, $c_a=0.1$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

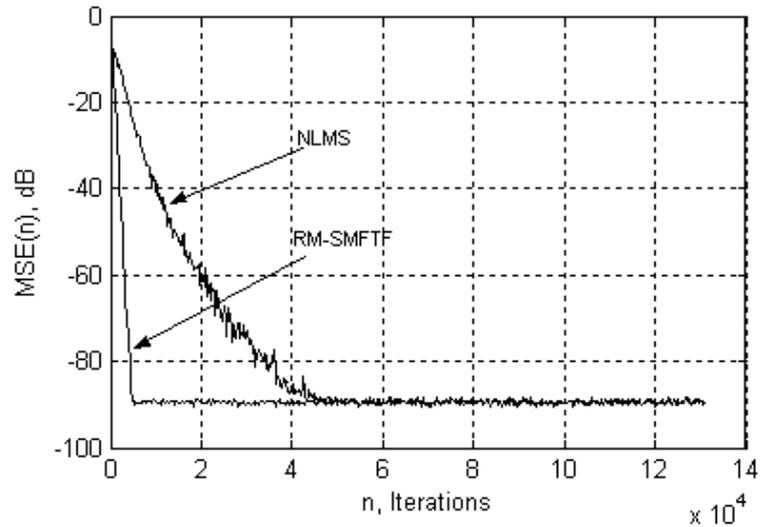


Fig.V.7: Comparaison entre les algorithmes RM-SMFTF et NLMS pour un canal stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie
 RM-SMFTF: $P=32$, $\lambda =0.9688$, $\eta =0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu =1$.

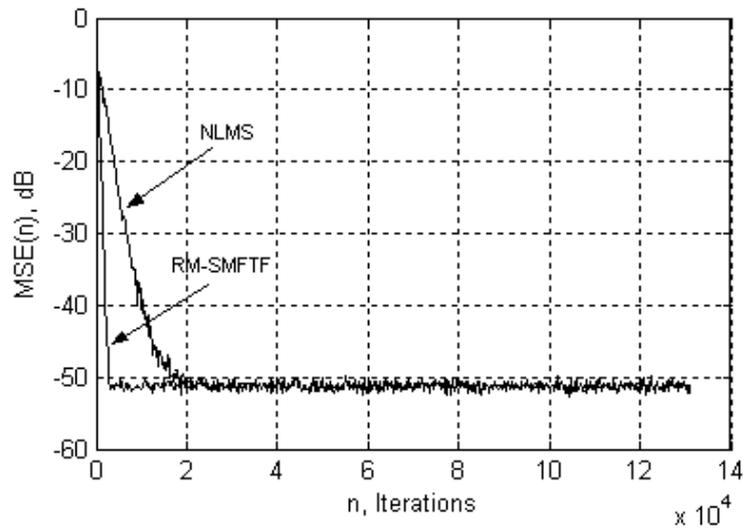


Fig.V.8: Comparaison entre les algorithmes RM-SMFTF et NLMS pour un canal stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, avec bruit additif sur la sortie ($SNR=50dB$).
 RM-SMFTF: $P=32$, $\lambda =0.9688$, $\eta =0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu =1$.

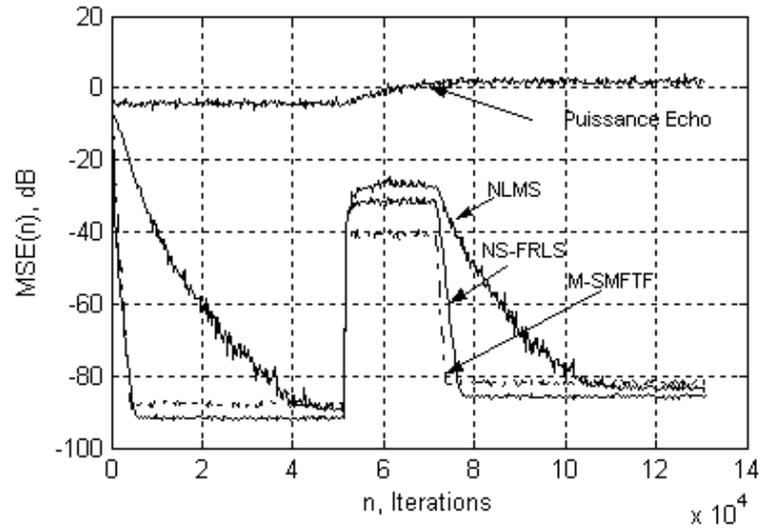


Fig.V.9: Evolution du *MSE* des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie.
M-SMFTF: $\lambda=0.9961$, $\eta=0.985$, $c_a=0.5$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

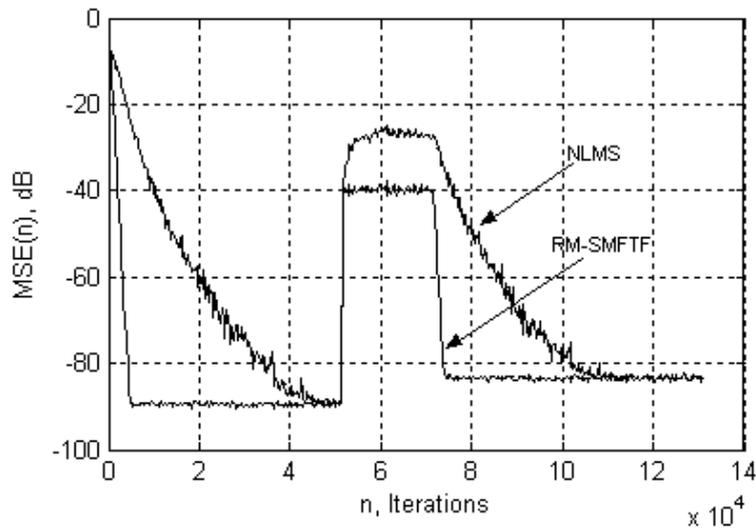


Fig.V.10: Evolution du *MSE* des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie.
RM-SMFTF: $P=32$, $\lambda=0.9688$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

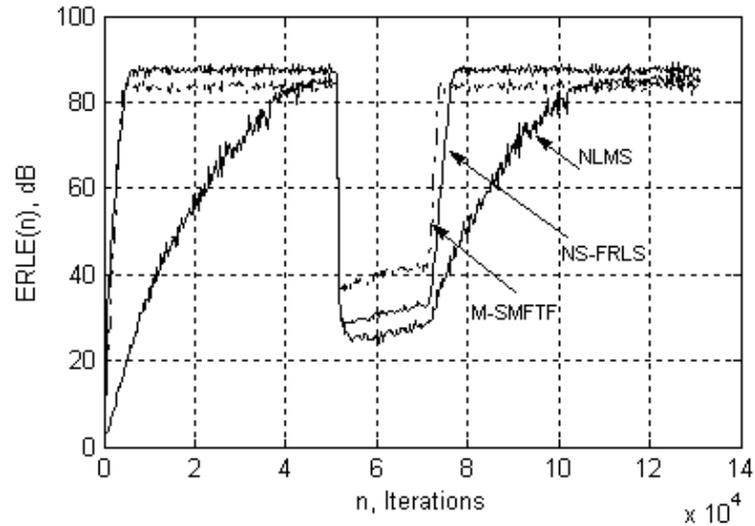


Fig.V.11: Evolution du $ERLE$ des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie.
M-SMFTF: $\lambda=0.9961$, $\eta=0.985$, $c_a=0.5$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

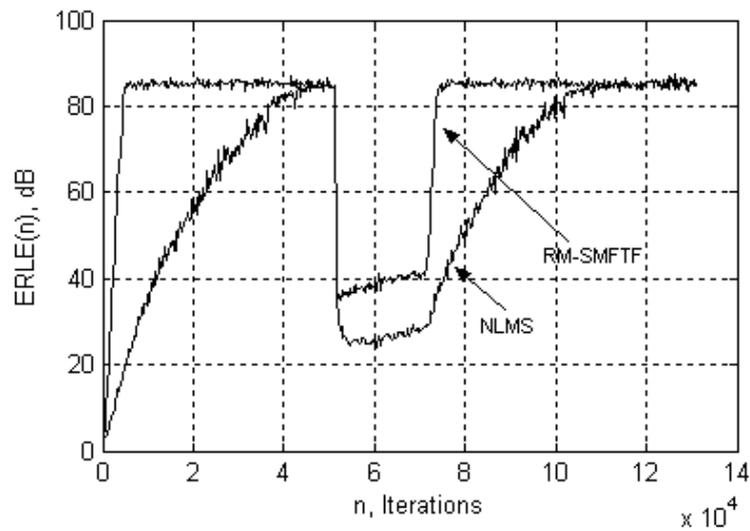


Fig.V.12: Evolution du $ERLE$ des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitable de voiture), Signal d'entrée stationnaire (USASI), Taille du filtre $L=256$, sans bruit additif sur la sortie.
RM-SMFTF: $P=32$, $\lambda=0.9688$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

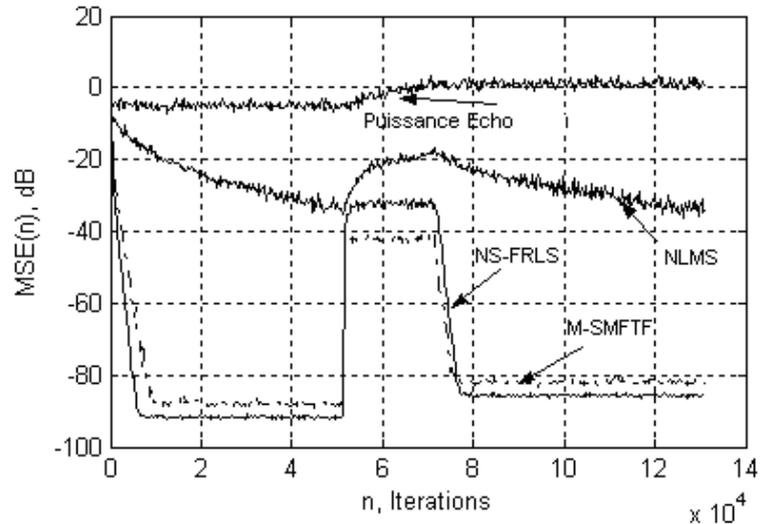


Fig.V.13: Evolution du *MSE* des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie.
 M-SMFTF: $\lambda=0.9961$, $\eta=0.996$, $c_a=0.5$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

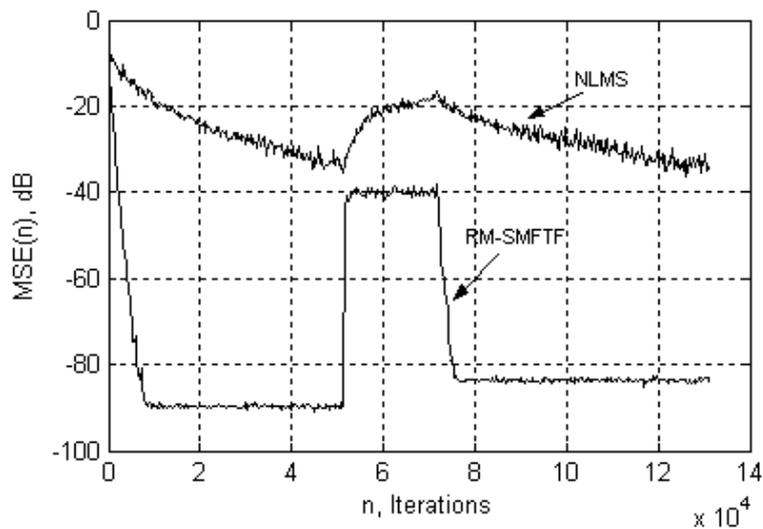


Fig.V.14: Evolution du *MSE* des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie.
 RM-SMFTF: $P=32$, $\lambda=0.9688$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

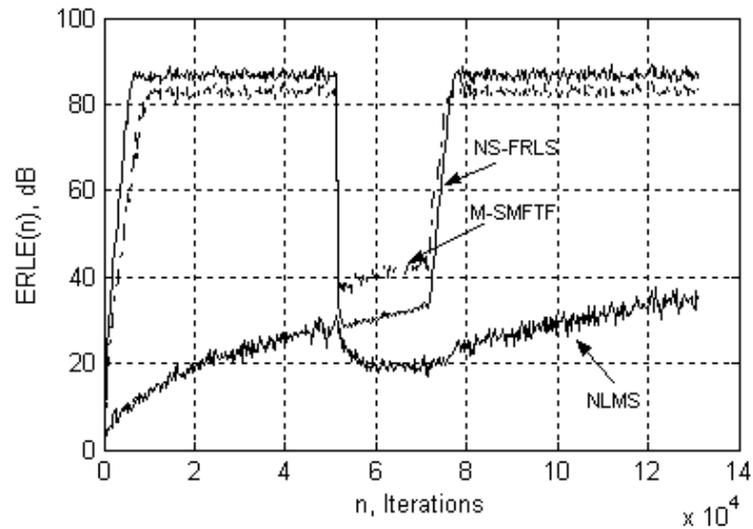


Fig.V.15: Evolution de $ERLE$ des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie.
M-SMFTF: $\lambda=0.9961$, $\eta=0.996$, $c_a=0.5$, $E_0=1$; NS-FRLS: $\lambda=0.9987$; NLMS: $\mu=1$.

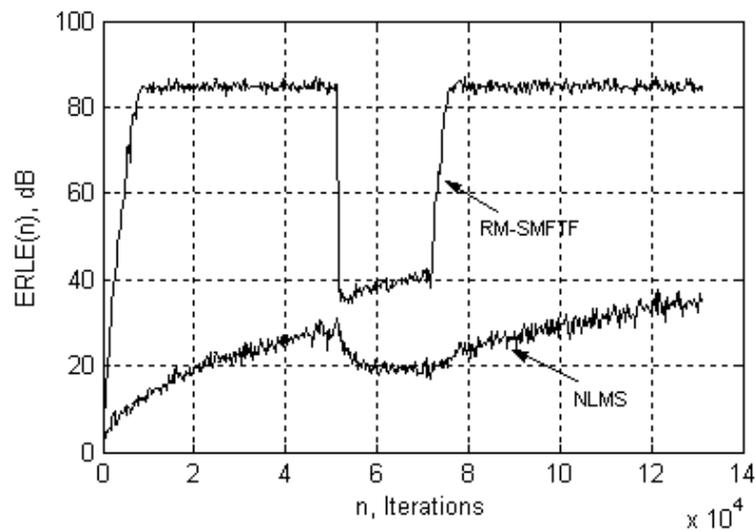


Fig.V.16: Evolution de $ERLE$ des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Habitacle de voiture), Signal d'entrée stationnaire (AR20), Taille du filtre $L=256$, sans bruit additif sur la sortie.
RM-SMFTF: $P=32$, $\lambda=0.9688$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

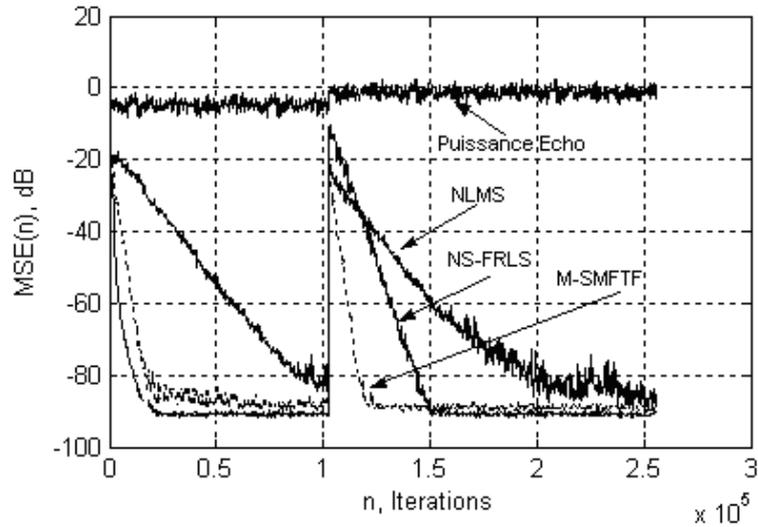


Fig.V.17: Evolution du *MSE* des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.

M-SMFTF: $\lambda=0.9993$, $\eta=0.99$, $c_a=0.5$, $E_0=5$; NS-FRLS: $\lambda=0.9998$; NLMS: $\mu=1$.

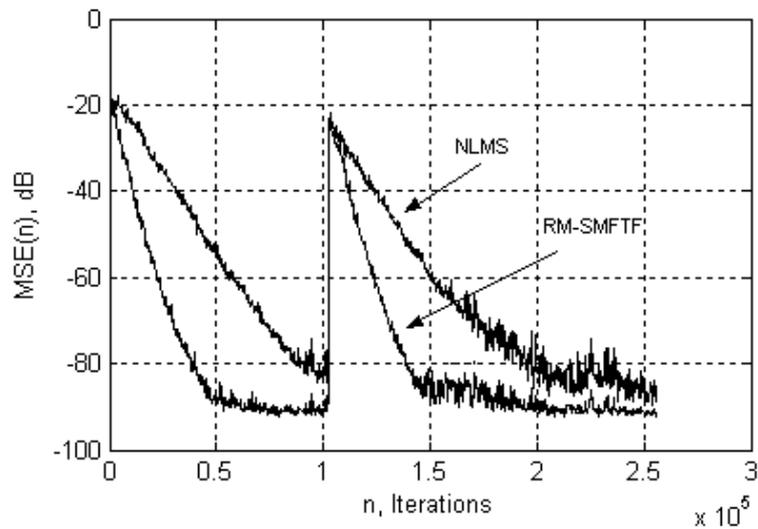


Fig.V.18: Evolution du *MSE* des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.

RM-SMFTF: $P=128$, $\lambda=0.9922$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

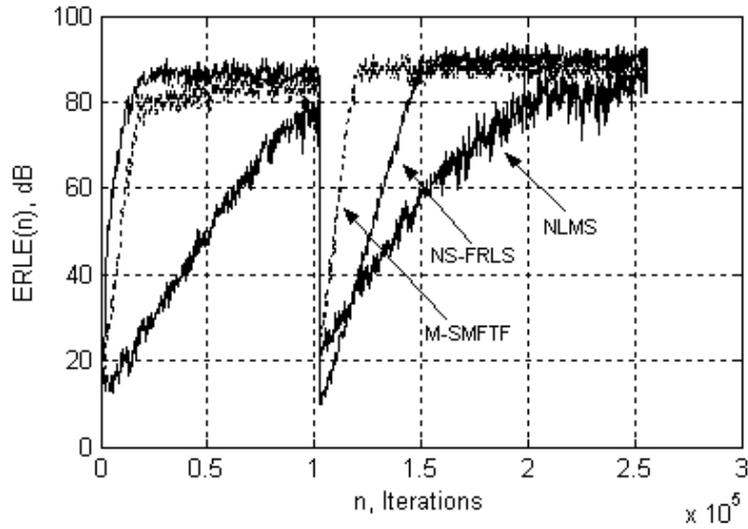


Fig.V.19: Evolution de $ERLE$ des algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.
M-SMFTF: $\lambda=0.9993$, $\eta=0.99$, $c_a=0.5$, $E_0=5$; NS-FRLS: $\lambda=0.9998$; NLMS: $\mu=1$.

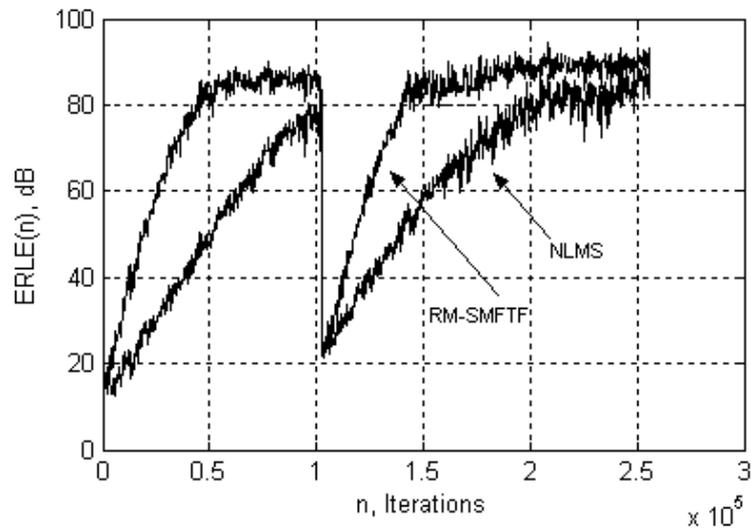


Fig.V.20: Evolution de $ERLE$ des algorithmes RM-SMFTF et NLMS pour un canal non stationnaire (Salle Téléconférence), Signal d'entrée stationnaire (USASI), Taille du filtre $L=1500$, sans bruit additif sur la sortie.
RM-SMFTF: $P=128$, $\lambda=0.9922$, $\eta=0.9985$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

V.5.3 Canal stationnaire avec entrée non stationnaire

Dans cette partie, nous présentons des simulations où la non stationnarité provient du signal d'entrée, le système est excité par le signal de parole [98]. Pour un filtre de taille L , le facteur d'oubli de l'algorithme NS-FRLS est $\lambda = 1 - 1/10L$ pour assurer la stabilité numérique et pour l'algorithme M-SMFTF, il reste inchangé et il est égale à $\lambda = 1 - 1/L$. La même remarque pour l'algorithme RM-SMFTF d'ordre P , le facteur d'oubli est $\lambda = 1 - 1/P$. La Fig.V.21 représente les performances adaptatives des algorithmes M-SMFTF, NS-FRLS et NLMS lorsque le système est les 256 coefficients de réponse impulsionnelle de l'habitacle de voiture. On observe que l'algorithme M-SMFTF réalise de meilleures performances que le NLMS. On remarque aussi que l'erreur résiduelle de l'algorithme M-SMFTF semble supérieure à celle du NS-FRLS. La simulation de la Fig.V.22, montre les performances adaptatives de l'algorithme M-SMFTF dans la même situation que la Fig.V.21 quand il y a un bruit blanc additif sur la sortie avec un rapport signal à bruit $SNR=50$ dB. On remarque que même dans le cas où le bruit est présent en sortie, l'algorithme M-SMFTF permet un plus grand choix de compromis entre la vitesse de convergence et l'erreur résiduelle que l'algorithme NLMS.

Les Fig.V.23 et Fig.V.24 représentent les performances adaptatives des algorithmes M-SMFTF, NS-FRLS, NLMS et RM-SMFTF lorsque le système est les 1500 coefficients de réponse impulsionnelle d'une salle de téléconférence. On observe que l'algorithme M-SMFTF réalise de meilleures performances que le NLMS, tant du point de vue de la vitesse de convergence que de l'erreur résiduelle. Il a une convergence comparable à celle du NS-FRLS. On voit aussi que l'algorithme RM-SMFTF converge plus vite que l'algorithme NLMS.

Nous avons simulé un changement brusque de l'écho au 56320^{ième} échantillon. Dans Fig.V.25, on peut voir que la convergence initiale est presque la même pour les algorithmes M-SMFTF et NSFRLS. Mais le M-SMFTF réalise une meilleure re-convergence après le changement brusque de la réponse impulsionnelle. On voit que l'algorithme RM-SMFTF (Fig.V.26) réalise de meilleures performances que l'algorithme NLMS. On remarque aussi que l'écho résiduel après convergence est toujours au dessous de l'écho.

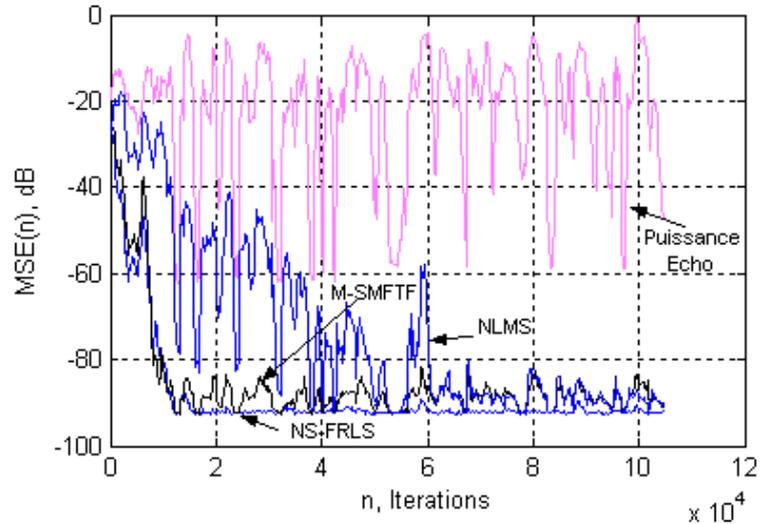


Fig.V.21: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.

M-SMFTF: $\lambda = 0.9961$, $\eta = 0.96$, $c_a = 0.1$, $E_0 = 0.5$; NS-FRLS: $\lambda = 0.9996$; NLMS: $\mu = 1$.

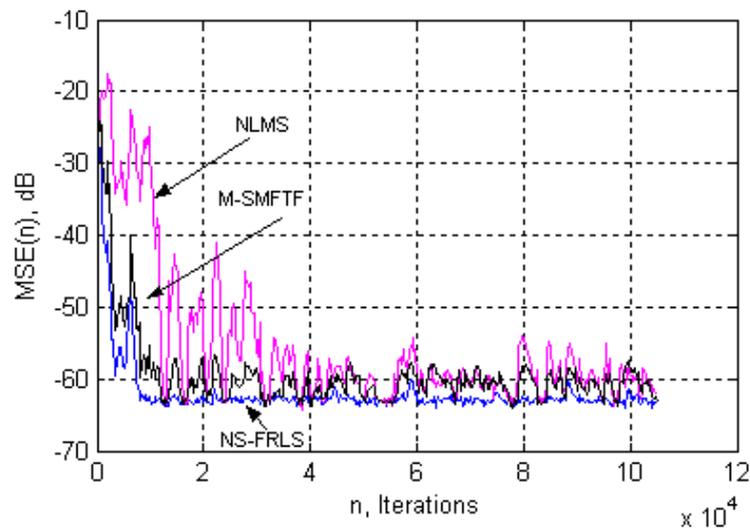


Fig.V.22: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, avec bruit additif sur la sortie ($SNR=50dB$).

M-SMFTF: $\lambda = 0.9961$, $\eta = 0.96$, $c_a = 0.1$, $E_0 = 0.5$; NS-FRLS: $\lambda = 0.9996$; NLMS: $\mu = 1$.

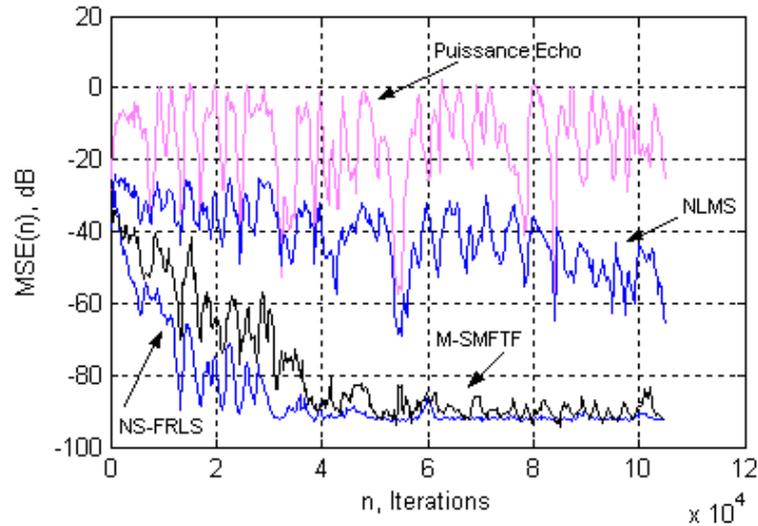


Fig.V.23: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal stationnaire (Salle Téléconférence), Entrée non stationnaire (Parole), Taille du filtre $L=1500$, sans bruit additif sur la sortie.

M-SMFTF: $\lambda=0.9993$, $\eta=0.96$, $c_a=0.2$, $E_0=3$; NS-FRLS: $\lambda=0.9999$; NLMS: $\mu=1$.

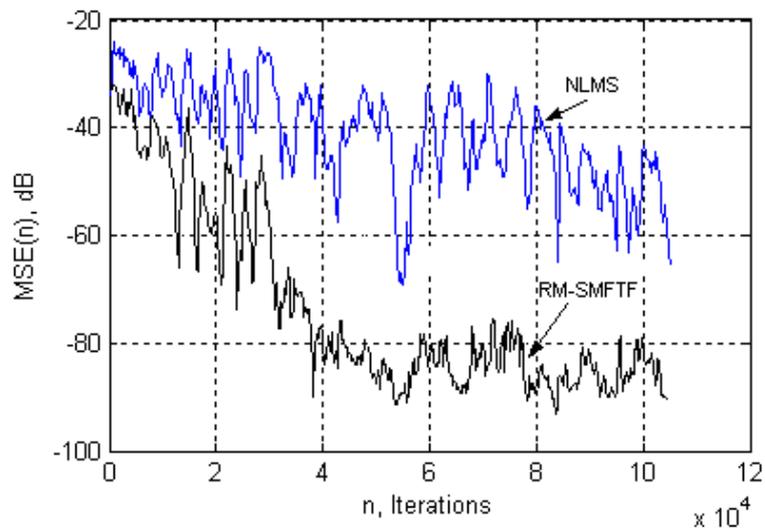


Fig.V.24: Comparaison entre les algorithmes RM-SMFTF NLMS pour un canal non stationnaire (Salle Téléconférence), Entrée non stationnaire (Parole), Taille du filtre $L=1500$, sans bruit additif sur la sortie.

RM-SMFTF: $P=128$, $\lambda=0.9922$, $\eta=0.99$, $c_a=0.5$, $E_0=0.2$; NLMS: $\mu=1$.

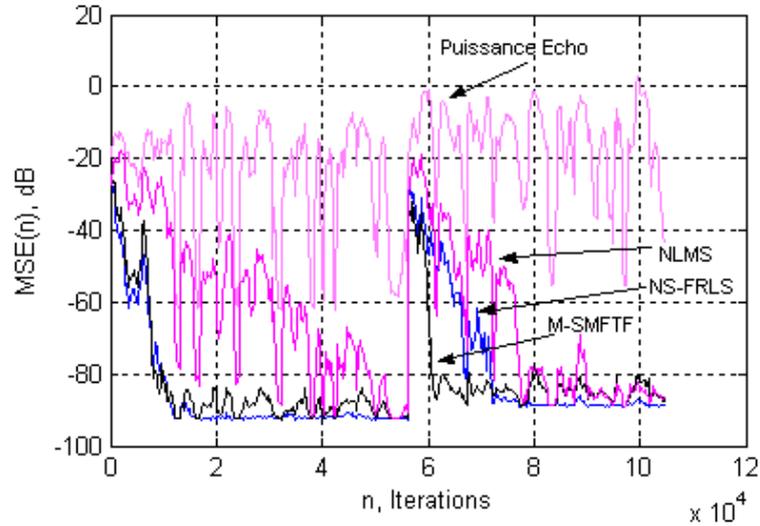


Fig.V.25: Comparaison entre les algorithmes M-SMFTF, NS-FRLS et NLMS pour un canal non stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.
M-SMFTF: $\lambda=0.9961$, $\eta=0.96$, $c_a=0.1$, $E_0=0.5$; NS-FRLS: $\lambda=0.9996$; NLMS: $\mu=1$.

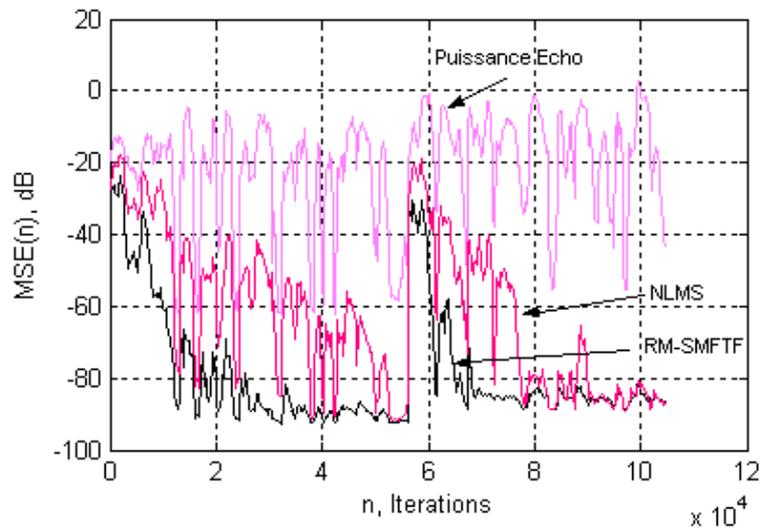


Fig.V.26: Comparaison entre les algorithmes RM-SMFTF NLMS pour un canal non stationnaire (Habitacle de voiture), Entrée non stationnaire (Parole), Taille du filtre $L=256$, sans bruit additif sur la sortie.
RM-SMFTF: $P=20$, $\lambda=0.950$, $\eta=0.99$, $c_a=0.1$, $E_0=1$; NLMS: $\mu=1$.

V.6 Conclusion

Dans ce chapitre, nous avons comparé les performances adaptatives de l'algorithme M-SMFTF à celles du NS-FRLS et NLMS dans plusieurs situations. Les premières simulations effectuées lorsque le signal d'excitation était stationnaire, très corrélé et le système à identifier était non stationnaire, ont permis de comparer la vitesse de convergence et la capacité de poursuite de l'algorithme M-SMFTF à celles de NS-FRLS et NLMS. Nous avons observé que l'algorithme proposé poursuit mieux que l'algorithme NS-FRLS. Nous avons évalué, dans les secondes simulations, les performances adaptatives de l'algorithme dans des situations réelles où le signal d'entrée était de la parole avec un bruit additif en sortie. En comparant ces performances adaptatives, nous avons constaté que l'algorithme M-SMFTF converge plus rapidement que le NLMS et poursuit légèrement mieux que le NS-FRLS. A partir de ces différents critères de performances, mesurés par l'erreur quadratique moyenne (MSE) et le rapport de la puissance de l'écho à celle de l'écho résiduel ($ERLE$), nous constatons que les algorithmes proposés M-SMFTF et RM-SMFTF fournissent de meilleurs résultats au sens de rendre l'écho résiduel moins audible en sortie du système d'annulation d'écho par rapport aux autres algorithmes NLMS et NS-FRLS.

CONCLUSION GENERALE

Nous avons posé le problème de l'annulation d'écho acoustique et sa nécessité dans les communications avec les postes mains-libres. En effet, le problème de l'écho acoustique est classiquement résolu à l'aide d'un annulateur d'écho en identifiant progressivement la réponse impulsionnelle finie d'un filtre adaptatif à partir des algorithmes de type moindres carrés. Pour cela, nous avons développé les équations issues de la structure transversale. L'identification du système commence par l'établissement de l'erreur entre la sortie du système et du filtre adaptatif. Par la suite, le choix d'un critère d'erreur nous permet de définir une fonction de coût que nous avons minimisé. Nous avons déduit alors l'équation de Wiener qui répond au problème de minimisation du critère d'erreur choisi. Nous avons rappelé les principaux résultats du LMS et ses versions. Ensuite, nous nous sommes attardés aux algorithmes des moindres carrés récursifs rapides (FRLS). Nous avons vu que la complexité de calcul du FRLS augmente linéairement avec la taille du filtre adaptatif d'ordre L comme dans le cas de l'algorithme LMS. Ainsi, l'algorithme FRLS offre le meilleur des deux mondes: la propriété de convergence de l'algorithme RLS et la simplicité de calcul de l'algorithme LMS.

Malheureusement l'algorithme FRLS est exponentiellement instable. Une certaine redondance nous a permis de combattre et d'avoir un contrôle sur l'erreur générée par l'accumulation et la propagation des erreurs numériques. Nous avons calculé les équations de propagation des erreurs dans l'algorithme FRLS, puis modifié par une contre réaction convenable, les propriétés numériques de l'algorithme. Nous avons analysé les propriétés numériques de l'algorithme FRLS stabilisé en utilisant un modèle du premier ordre de la propagation des erreurs numériques. Nous avons présenté de nouvelles versions numériquement stables des algorithmes des moindres carrés récursifs rapides (NS-FRLS). La condition de stabilité s'est avérée capable de maintenir une bonne performance de convergence par des simulations sur ordinateur. Ces algorithmes ainsi modifiés sont stables numériquement pour un choix convenable de certains paramètres ainsi d'une condition sur le facteur d'oubli. Ces nouvelles méthodes augmentent faiblement la complexité de calcul des algorithmes initiaux de L opérations supplémentaire pour chaque

méthode. L'introduction d'une nouvelle variable $NGE(n)$, nous a permis de comparer et montrer l'évolution de la stabilité des algorithmes. Nous avons observé que les résultats sont en accord avec les expressions théoriques. La stabilité numérique a été vérifiée par simulation sur de très grandes durées de signal stationnaire et pour différents ordres de filtre. Par la suite, nous avons présenté une étude sur le comportement des algorithmes NS-FRLS avec des sinusoïdes pures. D'une part, nous avons rassemblé un certain nombre de résultats sur la position des pôles et la divergence de l'algorithme. Pour stabiliser l'algorithme, nous avons proposé un choix de la fréquence d'échantillonnage par rapport à la fréquence de la sinusoïde. D'autre part, nous avons montré la possibilité d'avoir des algorithmes stables en fonctionnement continu avec un bruit additif. Si le bruit est nul, le signal d'entrée est un signal prédictible et les coefficients de prédiction sont tels que les pôles soient sur le cercle unité. Au fur et à mesure que le niveau du bruit augmente, les coefficients varient de sorte que les pôles soient ramenés vers l'intérieur du cercle.

L'algorithme NS-FRLS, converge rapidement, avec une bonne capacité de poursuite et une bonne robustesse au bruit ambiant, alors que sa complexité est très élevée ($8L$) et sa tendance à l'instabilité lorsque le signal d'entrée est de la parole. Pour cela, nous avons proposé un nouvel algorithme rapide (M-SMFTF) pour le filtrage adaptatif. Cet algorithme possède une complexité de calcul égale à $6L$. Cette complexité de calcul a été réduite à $(2L+4P)$ par l'algorithme RM-SMFTF en utilisant un prédicteur aller réduit d'ordre P , où P est très inférieur à L . Nous avons décrit les différentes étapes pour l'obtention de l'algorithme M-SMFTF. Nous avons proposé la réduction de l'ordre du prédicteur aller par une méthode d'extrapolation qui consistait à construire la matrice d'autocorrélation d'ordre L à partir d'une estimation de la matrice d'autocorrélation d'ordre P . Nous avons étudié la propagation des erreurs de toutes les quantités récursives de la partie prédiction de l'algorithme proposé. Nous avons analysé les propriétés numériques de l'algorithme. Cet algorithme est stable numériquement pour un choix convenable du facteur de fuite et une condition sur le facteur d'oubli. Ces analyses ont montré la supériorité en termes de capacité de poursuite de cet algorithme par rapport à l'algorithme NS-FRLS. Nous avons comparé, par simulations, les performances adaptatives de l'algorithme proposé à celles du NS-FRLS et du NLMS dans plusieurs situations. Les simulations ont prouvé la supériorité de l'algorithme proposé.

Nous avons comparé les performances adaptatives de l'algorithme M-SMFTF à celles du NS-FRLS et NLMS dans plusieurs situations. Les premières simulations effectuées lorsque le signal

d'excitation était stationnaire, très corrélé et le système à identifier était non stationnaire, ont permis de comparer la vitesse de convergence et la capacité de poursuite de l'algorithme M-SMFTF à celles de NS-FRLS et NLMS. Nous avons observé que l'algorithme proposé poursuit mieux que l'algorithme NS-FRLS. Nous avons évalué, les performances adaptatives de l'algorithme dans des situations réelles où le signal d'entrée était de la parole avec un bruit additif en sortie. En comparant ces performances adaptatives, nous avons constaté que l'algorithme M-SMFTF converge plus rapidement que le NLMS et poursuit légèrement mieux que le NS-FRLS. A partir de ces différents critères de performances, mesurés par l'erreur quadratique moyenne et le rapport de la puissance de l'écho à celle de l'écho résiduel, nous avons constaté que les algorithmes proposés M-SMFTF et RM-SMFTF fournissent de meilleurs résultats au sens de rendre l'écho résiduel moins audible en sortie du système d'annulation d'écho par rapport aux autres algorithmes NLMS et NS-FRLS.

Nous avons vu que les algorithmes adaptatifs proposés peuvent fournir de bons résultats au niveau de l'annulation d'écho acoustique lorsqu'une seule voix est émise. Cependant si deux personnes émettent simultanément un signal, cas de la double parole, l'algorithme d'adaptation ne va plus tendre vers une modélisation du chemin d'écho mais il aura tendance à diverger. En perspectives, nous pensons que la combinaison « d'un annulateur d'écho à un système de variation de gain » peut permettre d'éliminer complètement l'écho sans engendrer de dégradation sur le signal de parole locale. Et envisager une implantation de ce système combiné dans un microprocesseur de traitement de signal.

Bibliographie

- [1] A.Gilloire and J.P.Jullien: "L'acoustique des salles dans les télécommunications", L'Echo des recherches, No.127, 1^{er} trimestre, 1987.
- [2] S.H.Jensen: "Acoustic Echo Canceller for Hands-free Mobile Radiotelephony", EUSIPCO, European Sig. Proc. Conf., Brussels, Aug. 24-27, 1992, pp.1629-1632.
- [3] K.Murano, S.Unagami and F.Amano: "Echo cancellation and applications", IEEE Communication Magazine, January 1990.
- [4] J. Benesty, T. Gansler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, editors. Advances in Network and Acoustic Echo Cancellation. Springer-Verlag, 2001.
- [5] J.Proakis, C. M.Rader, F.Ling, M.Moonen, I.K.Proudlar and C.L.Nikias: "Algorithms for Statistical", Signal Processing. Prentice Hall, Englewood Cliffs, New Jersey, 2002.
- [6] C. Gritton and D. Lin, "Echo cancellation algorithms," IEEE Trans. Acoust., Speech, Signal Processing, vol. 32, April 1984.
- [7] J.P.Julien : "Acoustique des salles, prise et restitution du son, analysées à partir de la notion de canal acoustique", Note technique CENT NT/LAA/TSS/128, Août 1982.
- [8] C.R.Johnson Jr: "Adaptive IIR Filtering-Current Results and Open Issues", IEEE Trans. Inform. Theory, Vol.30, No.2, March 1984, pp.237-250.
- [9] O.Muron and J. Sikorav: "Modelling of reverberators and audioconference rooms", Proc. ICASSP 86, Tokyo, Japan, April 7-11, 1986, pp.921-924.
- [10] T.Petillon: "Réduction de complexité des algorithmes des moindres carrés rapides- Application à l'annulation d'écho acoustique dans les radio-mobiles", Thèse de Doctorat, Université Rennes 1, 1992.
- [11] M.Mboup : "Identification adaptative par structures prédictive -Application à l'annulation d'écho acoustique", Thèse de Doctorat, Université de Paris 11 Orsay France, 1992.
- [12] M.Montazeri : "Une famille d'algorithmes adaptatifs comprenant les algorithmes NLMS et RLS - Application à l'annulation d'écho acoustique", Thèse de Doctorat, Université de Paris-sud U.F.R. Scientifique D'Orsay, 1994.
- [13] K.Maouche: "Algorithmes des moindres carrés récursifs doublement rapides- Application à l'identification de réponses impulsionnelles longues", Thèse de Doctorat, Ecole Nationale Supérieure des Télécommunications, 1996.
- [14] V.Turbin : "Combinaison du filtrage adaptatif et du filtrage optimal pour réaliser l'annulation de l'écho acoustique dans un contexte de téléconférence", Thèse de Doctorat, Université de Rennes 1, France, 1998.
- [15] F.Berthault : "Contribution à l'amélioration des systèmes d'annulation d'écho acoustique", Thèse de Doctorat, Université de Cergy Pontoise, 2000.

- [16] L.Ljung and T. Söderström: “Theory and Practice of Recursive Identification”, MIT Press, 1983
- [17] B.Farhang-Boroujeny: “Adaptive Filters - Theory and Applications”, Wiley & Sons, October 1998.
- [18] S.S.Narayan, A.M.Peterson and M.J.Narasimha: “Transform Domain LMS Algorithm”, IEEE Transactions on Acoustics, Speech and Signal Processing, vol.31, No.3, June 1983, pp. 609-615.
- [19] K.Murano, S.Unagami and F.Amano: “Echo Cancellation and Applications”, IEEE Communications Magazine, Jan 1990, pp.49-55.
- [20] A.Feur and E.Weinstein: “Convergence Analysis of LMS Filters with Uncorrelated Gaussian Data”, IEEE Transactions on Acoustics, Speech and Signal Processing, vol.33, February 1985, pp. 222-230.
- [21] T.Kailath: “A View of Three Decades of Linear Filtering Theory”, IEEE Transactions on Information Theory, vol. 20, No.2, March 1974, pp.146-181.
- [22] T.Kailath: “Equations of Wiener-Hopf Type in Filtering Theory and Related Applications”, Norbert Wiener Collected Works, edited by P.Masani, MIT Press, vol.3, 1982, pp.63-64.
- [23] B.Widrow et al.: “Adaptive Noise Cancelling: Principles and Applications”, Proceedings of the IEEE, vol.63, No.12, December 1975, pp.1692-1716.
- [24] B.Widrow et al.: “Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter”, Proceedings of the IEEE, vol.64, No.8, August 1976, pp.1151-1162.
- [25] S.Haykin: “Lessons on Adaptive Systems for Signal Processing, Communications, and Control”, IEEE Signal Processing Magazine, September 1999, pp.39-48.
- [26] G.O.Glentis, K.Berberidis and S.Theodoridis: “Efficient Least Squares Adaptive Algorithms for FIR Transversal Filtering”, IEEE Signal Processing Magazine, July 1999, pp. 13-41.
- [27] W.Shu and Y.Tianren: “Algorithm for linear Convolution using Number Theoretic Transforms”, Electronics Letters, vol.24, No.5, March 1988, pp.249-250.
- [28] B.Widrow and S.D.Stearns: “Adaptive Signal Processing”, Englewood Cliffs, N.J., Prentice-Hall, 1985.
- [29] J.B.Foley and F.M.Boland: “A Note on convergence analysis of LMS adaptive filters with Gaussian data”, IEEE Trans. ASSP, vol.36, No.7, pp.1087-1089, 1988.
- [30] S. Haykin : “Adaptive Filter Theory”, Englewood Cliffs, NJ: Prentice-Hall, 1985.
- [31] D.T.M. Slock: “On the convergence behaviour of the LMS and the NLMS algorithms”, IEEE Trans. Signal Processing, 1993, vol.42, pp.2811-2825
- [32] A.Gilloire, E.Moulines, D.Slock and P.Duhamel: “State of art in echo cancellation”, in A.R. Figuers-vidal, ed., Digital Signal processing in telecommunication, Springer Berlin, 1996, pp.45-91.
- [33] D.Falconer and L.Ljung: “Application of Fast Kalman Estimation to Adaptive Equalisation”, IEEE Trans. Comm., vol.26, No.10, Oct. 1978, pp.1439-1446.

- [34] G.Carayannis, D.Manolakis and N.Kalouptsidis: "A fast sequential algorithm for least squares filtering and prediction", IEEE Trans. ASSP, vol.31, No.6, 1983, pp.1394-1402.
- [35] J.M.Cioffi and T.Kailath: "Fast recursive least squares transversal filters for adaptive filtering", IEEE Trans. on ASSP, vol.32, No.2, Apr.1984, pp.304-337.
- [36] S.Ljung and L.Ljung: "Error propagation analysis of RLS adaptation algorithms", Automatica, vol.21, No.2, 1985, pp.157-167.
- [37] 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, pp.335-368.
- [38] T. Kailath: "Linear systems", Prentice Hall, Englewood Cliffs, NJ, 1980.
- [39] M.Bellanger: "Analyse des signaux et filtrage numérique adaptatif", Masson, CNET-ENST, Paris, 1989.
- [40] R.Alcantra, J.Prado and C.Gueguen: "Fixed point implementation of the fast Kalman algorithm using a tms 32010 microprocessor", In Proc. EUSIPCO 86 Conf., , The Hague, North Holland, Sept.1986 pp.1335-1338.
- [41] L.Ljung, M.Morf and D.Falconer: "Fast calculation of gain matrices for recursive estimation schemes", International J. Control, vol.27, 1978, pp.1-19.
- [42] F.Ling and J.G.Proakis: "Numerical accuracy and stability-two problems of adaptive estimation algorithms caused by roundoff error", Proc. ICASSP, San Diego, USA, Mar. 1984, pp.3031-3034.
- [43] E.Eleftheriou and D.Falconer: "Tracking Properties and Steady-State Performance of RLS Adaptive Filer Algorithms", IEEE Trans. ASSP, vol.34, No.5, Oct. 1986, pp.1097-1110.
- [44] P.Fabre and C.Gueguen: "Improvement of the FRLS Algorithms via Normalization – Comparative study", IEEE Trans. ASSP, vol.34, No.2, April 1986, pp.296-308.
- [45] J.L.Botto and G.V.Moustakides: "Stabilization of FRLS Transversal Filters for Adaptive Filtering", Proc. ICASSP 87 Conf., vol.1, Dallas, USA, April 1987, pp.403-407.
- [46] J.L.Botto : "Étude des algorithmes transversaux rapides, application à l'annulation acoustique pour l'audioconférence", Thèse Université de Rennes, 1986.
- [47] M.Bellanger: "Engineering aspects of fast least squares algorithms in transversal adaptive filters", Proc. ICASSP 87 Conf., Dallas, USA, April 1987, pp. 2149-2152.
- [48] A.Benallal : "Etude des algorithmes des moindres carrés transversaux rapides et application à l'identification de réponses impulsionnelles acoustiques", Thèse Université de Rennes 1, 1989.
- [49] D.T.M. Slock and T. Kailath: "Numerically stable FRLS transversal filters", Proc. ICASSP 88 Conf., New York, USA, April 1988, pp.1365-1 368.
- [50] D.T.M. Slock and T. Kailath: "Numerically stable fast transversal filters for RLS adaptive filtering", IEEE Trans. Signal Processing, vol.39, no.1, jan.1991, pp.92-113.
- [51] A.Benallal and A.Gilloire: "A new method to stabilize fast RLS algorithms based on first order model of propagation of numerical errors", Proc. ICASSP, New York, 1988, pp.1365-1368.

- [52] M.Arezki, A.Benallal, A.Guessoum, and D.Berkani.: “Three New Versions of Numerically Stable Fast Least Squares Algorithms (NS-FRLS) for Adaptive Filtering”, Proc. 4th international symposium on Communication Systems Networks and Digital Signal Processing, Newcastle, U.K, 2004, pp.528-532.
- [53] M.Arezki, F.Ykhlef, P. Meyrueis, and D.Berkani.: “Numerically Stable Fast Least Squares Algorithms in Adaptive Filtering”, Proc. 10th World Multi-Conference on Systemics Cybernetics and Informatics, Orlando, Florida, USA, 2006, pp.221-224.
- [54] M.Arezki, A.Benallal, P.Meyrueis, A.Guessoum and D.Berkani: “Analysis of fast recursive least squares algorithms for adaptive filtering”, International Journal of Circuits, Systems and Signal Processing, Issue 1, Vol.1, 2007, pp.85-91.
- [55] S. Haykin: “Adaptive Filter Theory”, 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
- [56] M.Arezki, A.Benallal, P.Meyrueis, A.Guessoum and D.Berkani: “Analysis of Fast RLS Algorithms for Adaptive Filtering”, Proc. 11th WSEAS International Conference on SYSTEMS, Crete Island, Greece, July 23-25, 2007, pp.474-481.
- [57] O.Macchi and M.Bellanger : “Le point sur le filtrage adaptatif transverse”, Proc. 11^{ème} Colloque GRETSI, Nice, 1987, 1G-4G.
- [58] M.Arezki, F.Ykhlef, A.Guessoum, P.Meyrueis, and D.Berkani: “Stability of the fast RLS algorithms with perfectly predictable signals”, IEEE International Conference on Industrial Technology ICIT2006, Dec.15-17, 2006, Mumbai, India, pp.698-703.
- [59] D.W.Lin: “On digital implementation of the fast Kalman algorithm”, IEEE Trans. ASSP, vol.32, No.5, Oct.1984, pp.998-1005.
- [60] A.Benallal, A.Gilloire : “Instabilité et stabilité des MCR transversaux rapides excité par la parole”, Proc. 12^{ème} GRETSI Conf., Juan-les-Pins, France, 1989, pp.509-512.
- [61] M.Arezki, A.Benallal, F.Ykhlef, A.Guessoum and D.Berkani, “New method of comparison NS-FRLS and NLMS Algorithms for the acoustic echo cancellation”, Proc. 4th international symposium on Communication Systems Networks and Digital Signal Processing, Newcastle, U.K, 2004, pp.528-532.
- [62] A.Benallal, A.Gilloire : “Improvement of tracking capability of the numerically stable fast RLS algorithms for adaptive filtering”, Proc. ICASSP, Glasgow, Scotland, May 23-26, 1989, pp.1031-1034.
- [63] A.H.Sayed: “Fundamentals of Adaptive Filtering”, John Wiley & Sons, New York, NJ, USA, 2003.
- [64] O.Macchi: “The Least Mean Squares Approach with Applications in Transmission”, New York, Wiley, 1995.
- [65] J.R.Treichler, C.R.Johnson, and M.G.Larimore: “Theory and Design of Adaptive Filter”, Prentice Hall, 2001.
- [66] S.Qiao: “Fast adaptive RLS algorithms: a generalized inverse approach and analysis”, Signal Processing, IEEE Transactions on, Vol.39, Issue 6, June, 1991, pp.1455 -1459.
- [67] I.D.Skidmore, and I.K.Proudler: “The KaGE RLS algorithm”, IEEE Transactions on Signal Processing, Dec. 2003, Vol.51, No.12, pp.3094-3104.
- [68] J.M.Cioffi: “Limited-precision effects in adaptive filtering,” IEEE Trans. Circuits and

- Systems, 1987, CAS-34, pp.821-833.
- [69] J.R.Bunch, R.C. Le Borne, and I.K.Proudlar: "Understanding the explosive divergence of the FTF algorithm", Proc. 14th International Conference on Digital Signal Processing, 2, July 1-3, 2002, pp.655-658.
- [70] D.T.MSlock: "The backward consistency concept and round-off error propagation dynamics in recursive least-squares algorithms", Optical Engineering, June 1992, Vol.31, pp.1153-1169.
- [71] P.A.Regalia: "Numerical stability issues in fast least-squares adaptation algorithms", Optical Engineering, June 1992, Vol.31, pp.1144 -1152.
- [72] P.A.Regalia: "Past input reconstruction in fast least-squares algorithms," IEEE Trans. Signal Processing, Sept. 1997, Vol.45, No.9, pp.2231-2240.
- [73] J.R.Bunch, R.C. Le Borne, and I.K.Proudlar: "Measuring and maintaining Consistency: A hybrid FTF algorithm", Int. Journal of Applied Math. and Comp. Sci., 2001, Vol.11, No.5, pp.1203-1216.
- [74] M.Arezki, A.Benallal, P.Meyrueis, A.Guessoum and D.Berkani: "Error Propagation Analysis of Fast Recursive Least Squares Algorithms". Proc. 9th IASTED International Conference on Signal and Image Processing, Honolulu, Hawaii, USA, August 20-22, 2007, pp.97-101.
- [75] L.Guo and L.Ljung : "Exponential Stability of General Tracking Algorithms", IEEE Trans. on Automatic Control, vol.40, No.8, August 1995, pp.1376-1387.
- [76] L.Guo and L.Ljung : "Performance Analysis of General Tracking Algorithms", IEEE Trans. on Automatic Control, vol.40, No.8, August 1995, pp. 1388-11402.
- [77] A.P.Liavas and P.A.Regalia: "On the Numerical Stability and Accuracy of the Conventional Recursive Least Squares Algorithm", IEEE Trans. on Signal Processing, vol.47, No.1, January 1999, pp.88-96.
- [78] A.Carini and E.Mumolo : "A Numerically Stable Fast RLS Algorithm for Adaptive Filtering and Prediction Based on the UD Factorization", IEEE Trans. on Signal Processing, vol.47, No. 8, August 1999, pp.2309-2313
- [79] S.Binde: "A numerically stable fast transversal filter with leakage correction", IEEE Signal Processing Letters, 1995, Vol.2, No.6, pp.114-116.
- [80] G.V.Moustakides and S.Theodoridis: "Fast Newton transversal filters - A new class of adaptive estimation algorithms", IEEE Trans. Signal Process, 1991, Vol.39, No.10, pp.2184-2193.
- [81] T.Petition, A.Gilloire and S.Theodorides: "The fast Newton transversal filter: An efficient scheme for acoustic echo cancellation in mobile radio", IEEE Trans. Signal Processing, March 1993, Vol.42, pp.509-518.
- [82] P.P.Mavridis, and G.V.Moustakides: "Simplified Newton-Type Adaptive Estimation Algorithms", IEEE Trans. Signal Process, 1996, Vol.44, No.8, pp.1932-1940.
- [83] W.Youhua, K.Ikeda and K.Nakayama: "A numerically stable fast Newton-type adaptive filter based on order recursive least squares algorithm", IEEE Transactions on Signal Processing, Sept. 2003, Vol.51, No.9, pp.2357-2368.

- [84] D.T.M.Slock and K.Maouche: “The fast subsampled-updating recursive least-squares (FSU RLS) algorithm for adaptive filtering based on displacement structure and the FFT”, *Signal Processing*, vol. 40, no. 2, pp.5–20, 1994.
- [85] X.-H.Yu, and Z.-Y.He: “Efficient Block Implementation of Exact Sequential Least Squares Problems”, *IEEE Trans. Acoust., Speech and Signal Proc.*, ASSP-36, March 1988, pp.392-399.
- [86] K.Maouche and D.T.M.Slock: “Fast Subsampled - Updating Stabilized Fast Transversal Filter (FSU SFTF) RLS algorithm for adaptive filtering”, *IEEE Transactions on Signal Processing*, Aug. 2000, Vol.48, No.8, pp.2248-2257.
- [87] K.Maouche, and D.T.M.Slock: “The Fast Subsampled-Updating Fast Newton Transversal Filter (FSU FNTF) Algorithm for Adaptive Filtering”, *IEEE Trans. On Circuits And Systems-II: Analog And Digital Signal Proc.*, Oct.2000, Vol.47, No.10, pp.1047-1058.
- [88] A.Benallal and A.Benkrid: “A simplified FTF-type algorithm for adaptive filtering”, *Signal processing*, 2007, Vol.87, No.5, pp.904-917.
- [89] M.Arezki, A.Benallal, A.Guessoum and D.Berkani: “Improvement of the Simplified FTF-type Algorithm”, *Proc. 5th International Conference on Signal Processing and Multimedia Applications, ICETE-SIGMAP 2008, Porto, Portugal, July 26–29, 2008*, pp.156-161.
- [90] M.Arezki and D.Berkani: “Advanced algorithms for adaptive filtering”, *Proc. of the 8th WSEAS International Conference on SIGNAL PROCESSING, ROBOTICS and AUTOMATION, February 21-23, 2009, Cambridge, UK*, pp. 113-119.
- [91] M. Arezki, A. Guessoum, P. Meyrueis: “Fast Algorithms for Adaptive Filtering”, the 16th IWSSIP International Conference on Systems, Signals and Image Processing, Greece, June 18-20, 2009, pp.1-4.
- [92] K.Mayyas: “Performance Analysis of the Deficient Length LMS Adaptive Algorithm”, *IEEE Trans. on Signal Processing*, vol.53, No.8, August 2005, pp.2727-2734.
- [93] A.Gilloire: “State of the art in acoustic echo cancellation”, in *adaptive algorithms – Applications and non classical schemes*, Publication da Universidade de Vigo, 1991, pp.20-31.
- [94] E.Hänsler: “The hands-free telephone problem – An annotated bibliography”, *Signal Processing*, vol.27, 1992, pp.259-271.
- [95] A.Gilloire: “recent advances in adaptive filtering algorithms for acoustic echo cancellation”, in *International Workshop on Acoustic Echo and Noise Control, Roros, Norway, June 1995*, pp.115-134.
- [96] P.Åhgren: “Acoustic Echo Cancellation and Doubletalk Detection Using Estimated Loudspeaker Impulse Responses”, *IEEE Trans. on Speech and Audio Processing*, vol.13, No.6, Nov. 2005, pp.1231-1237.
- [97] S.J.Park, C.G.Cho and D.H.Youn: “Integrated Echo and Noise Canceler for Hands-Free Applications”, *IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing*, vol.49, No.3, March 2002, pp.188-195.
- [98] M.Arezki A.Benallal, P.Meyrueis and D.Berkani: “A New Algorithm with Low Complexity for Adaptive Filtering”, *IAENG Journal, Engineering Letters*, vol.18, Issue 3, August 2010, pp.205-211.