

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي  
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البلدة  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الإلكترونيك  
Département d'Électronique



## Mémoire de Master

Mention Electronique  
Spécialité Système de Télécommunication

présenté par

KEDJAR Abdelhak

&

RAMDANE Mohamed Amine

# Algorithmes rapides pour l'identification de systèmes parcimonieux

Proposé par : Pr A. Benallal.

Année Universitaire 2017-2018



## **Abstract:**

---

Acoustic impulse response identification and echo cancellation are common problems in the field of signal processing. In this work, firstly, we present a theoretical detail of sparse algorithms which based on NLMS, FNLMS and RLS algorithms and used in the adaptive filter.

Finally, by using MATLAB® software, a series of simulations were carried out in the synthetic and real different sparseness acoustic impulse responses with stationary and non-stationary input in order to analyze, investigate and compare the algorithms strengths and weaknesses in terms of convergence speed, estimation accuracy.

Key word: adaptive filtering, sparse algorithms, system identification, acoustic impulse responses, acoustic echo cancellation.

## Remerciements

---

Avant tout, nous remercions Allah le tout puissant de nous avoir accordé la santé, le courage et les moyens pour suivre nos études et la volonté, la patience et la chance pour la réalisation de ce travail.

Nous sincères remerciements et notre profonde gratitude s'adressent à notre promoteur Pr : A. BENALLAL, pour avoir accepté de diriger ce travail, pour la grande patience, ses encouragements, ses orientations et ses conseils précieux jusqu'à dernière minute.

Notre gratitude à l'ensemble du personnel qui nous a motivé à la réalisation de Ce travaille pour leur gentillesse, leur disponibilité et leur aide.

## *Dédicace:*

Toutes les lettres ne sauraient trouver les mots  
qu'il faut est tous les mots ne sauraient exprimer  
la gratitude, le respect, la reconnaissance.

*Nous dédions ce mémoire aux :*

À nous chers parents aucune dédicace ne saurait  
exprimer mon respect, à nous amis en souvenir de  
notre sincère et profonde amitié et des moments  
agréables que nous avons passés ensemble.

Veuillez trouver dans ce travail l'expression de  
notre respect le plus profond et notre affection la  
plus sincère, à toutes les personnes qui ont  
participé à élaboration de ce travail.

## Listes des notations et abréviations :

### ➤ Notations :

$[.]^T$  Opération de transposition.

$[.]^{-1}$  Opération de matrice inverse.

$|. |$  Opération valeur d'absolue.

$E(.)$  Espérance mathématique.

$diag\{.\}$  Opération de diagonal d'une matrice.

$e^{(.)}$  Opération d'exponentiel.

$\ln(.)$  Fonction de logarithme.

$\log_{10}(.)$  Logarithme à base 10.

$\max\{.\}$  Opération de maximum.

$sgn(.)$  Fonction de signe.

$trace(.)$  Trace d'une matrice.

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

$\|.\|_1$  La norme  $l_1$ .

$\|.\|_2$  La norme  $l_2$ .

$\nabla$  Vecteur de gradient.

$\frac{\partial(.)}{\partial(.)}$  Dérivé partiel.

### ➤ Abréviations :

**EM** : Expectation **M**aximisation (Espérance et Maximisation).

**FIR** : Finite Impulse **R**esponse (Réponse Impulsionnelle finie).

**FNLMS** : **F**ast **N**ormalised **L**MS.

**HP** : hautparleur.

**IIR** : Infinite Impulse **R**esponse (Réponse Impulsionnelle Infinie).

**LMS** : Least **M**ean **S**quare (Moindre Carré Moyen)

**MIC** : Microphone.

**MSE** : **M**ean **S**quare **E**rror (Erreur Quadratique Moyen)

**NLMS** : **N**ormalised **L**MS (Moindre Carré Moyen Normalisé).

**RI** : **R**éponse Impulsionnelle.

**RID** : **R**éponse Impulsionnelle **D**ispersiv.

**RIP** : Réponse Impulsionnelle Parcimonieuse.

**RLS** : Recursive Least Square (Moindre Carré Récursif).

**RZA-FNLMS**: Reweighted ZA-FNLMS.

**RZA-NLMS**: Reweighted ZA-NLMS.

**SNR** : Signal to Noise Ratio (Rapport Signal sur Bruit).

**USASI** : USA Standard Institute.

**ZA-FNLMS**: Zero Attracting-FNLMS.

**ZA-NLMS** : Zero Attracting NLMS.

# Table des matières

Remerciements

Dédicaces

Résumés

Abréviations

*Introduction générale*----- 1

**Chapitre 1 Identification adaptative des réponses impulsionnelles (RI) acoustique**--- 3

1.1 Introduction----- 3

1.2 Réponse impulsionnelle (RI) acoustique d'une salle----- 3

1.3 Réponse impulsionnelle parcimonieuse----- 5

1.4 L'écho acoustique ----- 6

1.5 L'identification adaptative de RI----- 6

1.6 Annulation d'écho acoustique ----- 8

1.7 Définitions et formules----- 9

1.7.1 Le filtre optimum----- 10

1.7.2 Algorithme basé sur le gradient stochastique----- 11

a Algorithme de moindre carrée moyenne (LMS) ----- 11

b Algorithme de LMS normalisé (NLMS) ----- 12

1.7.3 Estimation des moindres carrées ----- 13

a Algorithme de moindres carrées récursifs (RLS)----- 13

1.8 Conclusion ----- 15

**Chapitre 2 Algorithmes d'identifications parcimonieux**----- 16

2.1 Introduction----- 16

2.2 Les algorithmes adaptatifs pour les RI parcimonieuses ----- 16

2.2.1 Algorithme zero attracting NLMS (ZA-NLMS)----- 16

2.2.2 Algorithme Reweighted Zero Attracting NLMS (RZA-NLMS) ----- 18

2.2.3 L'effet de  $\rho(R)ZA$  ( $\rho ZA$  ou  $\rho RZA$ ) ----- 19

2.3 Algorithme des moindres carrées rapide ----- 20

2.3.1 Algorithme Fast convergence NLMS (FNLMS)----- 20

2.4 Les algorithmes FNLMS parcimonieux ----- 25

2.4.1 Algorithme zero-attracting FNLMS (ZA-FNLMS) ----- 25

2.4.2 Algorithme Reweighted zero-attracting FNLMS (RZA-FNLMS) ----- 26

2.5 Algorithme d'Expectation-Maximisation RLS ----- 28



2.6	Conclusion	29
<b>Chapitre 3    <i>Résultat de simulations</i></b>		
3.1	Mesure du degré de parcimonie	30
3.1.1	La norme d'un vecteur	30
3.1.2	Mesure du degré de la parcimonie	31
3.2	Génération synthétique de RI parcimonieuse	32
3.3	Signaux de la simulation	32
3.4	Description des critères de performance :	34
3.5	Simulation des algorithmes	35
3.6	Simulation des algorithmes (R)ZA-NLMS et NLMS	35
a	Le choix de paramètre $\rho_{RZA}$ :	35
b	Le choix de paramètre $\epsilon_{RZA}$	36
3.6.2	Cas d'entrée stationnaire (USASI)	37
3.6.3	Cas d'entrée non-stationnaire (parole)	44
3.7	Simulation des algorithmes (R)ZA-FNLMS et FNLMS	48
3.7.1	Cas d'un signal stationnaire (USASI)	48
3.7.2	Cas d'un signal non-stationnaire (parole)	51
3.8	L'algorithme EM-RLS et RLS	54
3.9	Conclusion	56
<b>Conclusion générale</b>		<b>57</b>
<b>Bibliographie</b>		<b>58</b>

## Liste des figures

Figure1. 1: les réflexions de son dans une salle. ....	4
Figure1. 2: réponse impulsionnelle dispersive.....	5
Figure1. 3: réponse impulsionnelle parcimonieuse. ....	5
Figure1. 4: L'écho acoustique entre deux salles. ....	6
Figure1. 5: le concept d'indentification adaptative du système. ....	8
Figure1. 6: Principe d'annulation d'écho acoustique.....	9
Figure3. 1:RI synthétique avec $L=256$ , $Lp=30$ , $\psi=10,60,160$ . ....	33
Figure3. 2: RI réel avec $L=256$ , d'un habitacle de voiture, d'une salle audioconférence et d'une salle visio-conférence. ....	33
Figure3. 3: signal d'entrée stationnaire USASI. ....	34
Figure3. 4: signal d'entrée non-stationnaire (la parole).....	34
Figure3. 5: l'effet de $\rho RZA$ sur le comportement de l'algorithme ZA-NLMS, SNR=30.....	36
Figure3. 6: l'effet de $\varepsilon RZA$ sur le fonctionnement de l'algorithme RZA-NLMS.....	37
Figure3. 7: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	38
Figure3. 8: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=60$ , $\mu=1$ , SNR=30dB. ....	38
Figure3. 9: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=20dB. ....	39
Figure3. 10: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ...	40
Figure3. 11: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=50dB. ...	40
Figure3. 12: signal d'entrée USASI, RI synthétique (256) avec $L=128$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	41
Figure3. 13 signal d'entrée USASI, RI synthétique (256) avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	42
Figure3. 14: signal d'entrée USASI, RI synthétique (256) avec $L=350$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	42
Figure3. 15: signal d'entrée USASI, RI d'un habitacle d'une voiture (256) avec $L=256$ , $\mu=1$ , SNR=30dB. ....	43
Figure3. 16: signal d'entrée USASI, RI d'un habitacle d'une voiture (256) avec $L=350$ , $\mu=1$ , SNR=30dB. ....	43
Figure3. 17: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	44
Figure3. 18: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=256$ , $\psi=60$ , $\mu=1$ , SNR=30dB. ....	45
Figure3. 19: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=1024$ , $\psi=10$ , $\mu=1$ , SNR=30dB. ....	46
Figure3. 20 signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=1024$ , $\psi=10$ , $\mu=1$ , SNR=50dB. ....	46

Figure3. 21: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=1024$ , $\psi=60$ , $\mu=1$ , $SNR=30dB$ . .....	47
Figure3. 22: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=1024$ , $\psi=60$ , $\mu=1$ , $SNR=50dB$ . .....	47
Figure3. 23: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , $SNR=50dB$ . ...	49
Figure3. 24: signal d'entrée USASI, RI synthétique avec $L=256$ , $\psi=60$ , $\mu=1$ , $SNR=50dB$ . ...	49
Figure3. 25: signal d'entrée USASI, RI synthétique avec $L=1024$ , $\psi=10$ , $\mu=1$ , $SNR=30dB$ ..	50
Figure3. 26: signal d'entrée USASI, RI synthétique avec $L=1024$ , $\psi=60$ , $\mu=1$ , $SNR=30dB$ . .	50
Figure3. 27: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , $SNR=20dB$ . .....	51
Figure3. 28: signal d'entrée Parole, RI synthétique avec $L=256$ , $\psi=10$ , $\mu=1$ , $SNR=30dB$ . ..	52
Figure3. 29: signal d'entrée non-stationnaire (Parole), RI synthétique (512) avec $L=256$ , $\psi=10$ , $\mu=1$ , $SNR=50dB$ . .....	52
Figure3. 30 signal d'entrée non-stationnaire (Parole), RI synthétique (512) avec $L=350$ , $\psi=10$ , $\mu=1$ , $SNR=50dB$ . .....	53
Figure3. 31: signal d'entrée non-stationnaire (Parole), RI synthétique avec $L=750$ , $\psi=10$ , $\mu=1$ , $SNR=50dB$ . .....	53
Figure3. 32: Signal d'entrée stationnaire (USASI), RI synthétique avec $L=256$ , $\psi=10$ , $\lambda=0.995$ , $\delta=0.33$ , $SNR=25dB$ . .....	54

## Liste des tableaux

Table 1. 1: L'algorithme LMS. ....	12
Table 1. 2: L'algorithme NLMS.....	13
Table 1. 3: L'algorithme RLS. ....	15
Table 2. 1: L'algorithme ZA-NLMS. ....	18
Table 2. 2: L'algorithme RZA-NLMS.....	19
Table 2. 3: L'algorithme FNLMS.....	24
Table 2. 4: L'algorithme ZA-FNLMS. ....	26
Table 2. 5: L'algorithme RZA-FNLMS. ....	27
Table 2. 6: L'algorithme EM-RLS.....	29

# Introduction générale

---

Le filtrage adaptatif a été largement utilisé dans le cadre de nombreux domaines, notamment les télécommunications, le traitement du signal géophysique, le traitement du signal biomédical, l'élimination du fouillis radar et le traitement par sonar. Dans le domaine des communications numériques, les approches de filtrage adaptatif sont très importantes et ont été appliquées dans de nombreuses applications telles que l'annulation de bruit, la formation de faisceau adaptatif et l'égalisation du canal.

Une application spécifique de l'annulation de bruit, qui devient de plus en plus répandue dans les télécommunications, est l'annulation d'écho acoustique. Plusieurs algorithmes de filtrage adaptatifs ont été proposés et étudiés pour surmonter les problèmes d'écho dans les réseaux mobiles et de communication mains-libres, les systèmes audio et visioconférence.

Le but de ce projet est l'étude d'algorithmes adaptatifs rapides et simplifiés qui tiennent compte du caractère parcimonieux des réponses impulsionnelles acoustiques et leurs applications dans l'annulation d'écho acoustique.

Notre mémoire est organisée en trois chapitres comme suit :

Dans le premier chapitre nous allons définir les réponses impulsionnelles acoustiques parcimonieuses ou non dans le contexte d'une application d'annulation d'écho acoustique adaptative. Nous présentons aussi le concept du filtre optimal de Wiener et les deux grandes familles d'algorithmes d'identification adaptative : algorithmes de type gradient stochastique (LMS) ou des algorithmes de type moindres carrés récursifs (RLS).

Dans le deuxième chapitre, nous allons discuter une famille d'algorithmes de type gradient stochastique NLMS pour l'identification de réponses impulsionnelles parcimonieuses ;

notre intérêt c'est porté sur la catégorie d'algorithmes zero attracting NLMS et ses variantes, puis nous proposons une extension de ces principes de parcimonie au cas de l'algorithme fast NLMS. Nous avons aussi abordé l'algorithme RLS parcimonial basé sur le principe Expectation-Maximisation (EM).

Dans le troisième chapitre nous présentons des résultats de simulations qui comparent les performances des algorithmes adaptatifs présentés dans les chapitres précédents, ces performances sont basées sur la vitesse de convergence et l'énergie de l'erreur de filtrage final.

Enfin, notre mémoire sera terminé par une conclusion générale.

# Chapitre 1 Identification adaptative des réponses impulsionnelles (RI) acoustique

---

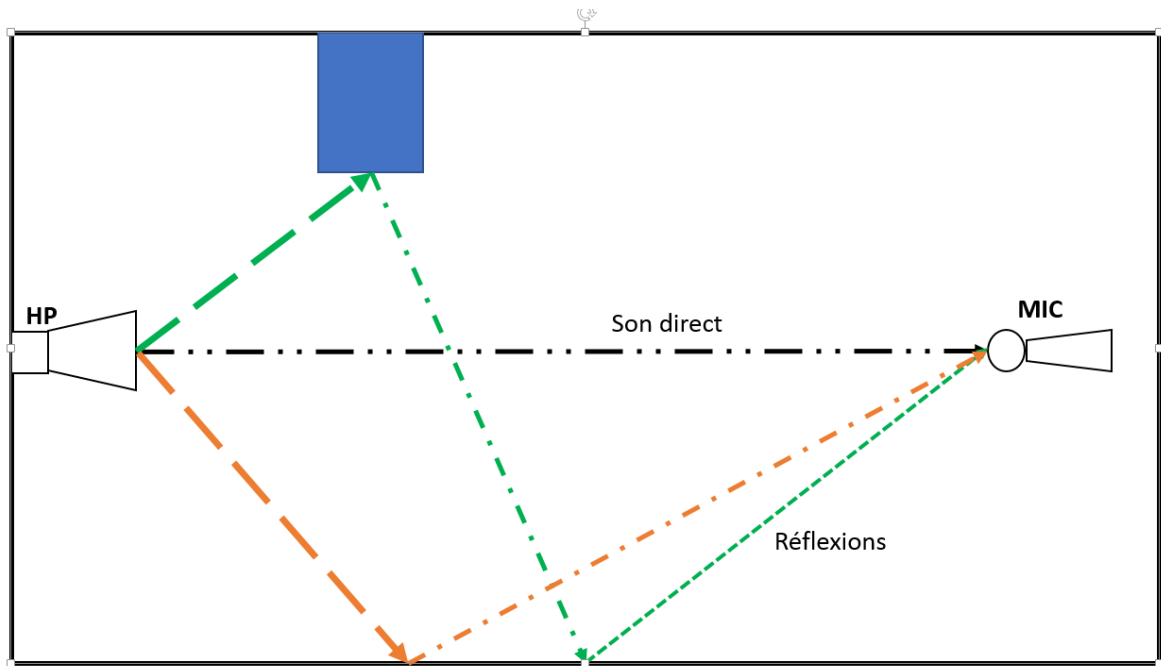
## 1.1 Introduction

Une application spécifique au filtrage adaptatif, qui devient de plus en plus répandue dans les télécommunications, est l'annulation adaptative de l'écho acoustique. Plusieurs algorithmes de filtrage adaptatif ont été proposés et étudiés pour surmonter les problèmes d'écho acoustique dans plusieurs applications réelles (téléphone mains libres, téléconférence, vidéoconférence, habitacle de voiture ... etc.).

Les difficultés essentielles rencontrées sont la durée importante des réponses impulsionnelles des canaux acoustiques à identifier et la nature des signaux à traiter, cela va nous conduire dans ce chapitre à définir les réponses impulsionnelles acoustique en général et parcimonieuses en particulier ensuite nous allons définir l'écho acoustique avec son influence sur notre information et nous expliquons les solutions pour traiter ce problème, à la fin nous donnons les grand types des algorithmes adaptatifs classiques et leurs variantes.

## 1.2 Réponse impulsionnelle (RI) acoustique d'une salle

Pour la définition de la réponse impulsionnelle acoustique on place un témoin (récepteur) dans une salle vide, et on lance un morceau de son à partir d'un média équipé par des haut-parleurs (source), le récepteur capte le son direct et plusieurs réflexions des parois de la salle.



*Figure1. 1: les réflexions de son dans une salle.*

La figure (1.1) présente la propagation de son dans une salle contenant quelques objets, chaque réflexion est retardée et atténué. Le microphone placé loin du haut-parleur capte les différents sons qui se produisent par ces réflexions. Dans la théorie de l'acoustique linéaire, ces réflexions sont modélisées par un filtre à réponse impulsionnelle finie (RIF). En pratique, selon les dimensions de la salle, la taille de RI varie de quelques certains de coefficient à plusieurs milliers de coefficients, La figure (1.2) montre un exemple de la réponse impulsionnelle dispersive (RID).



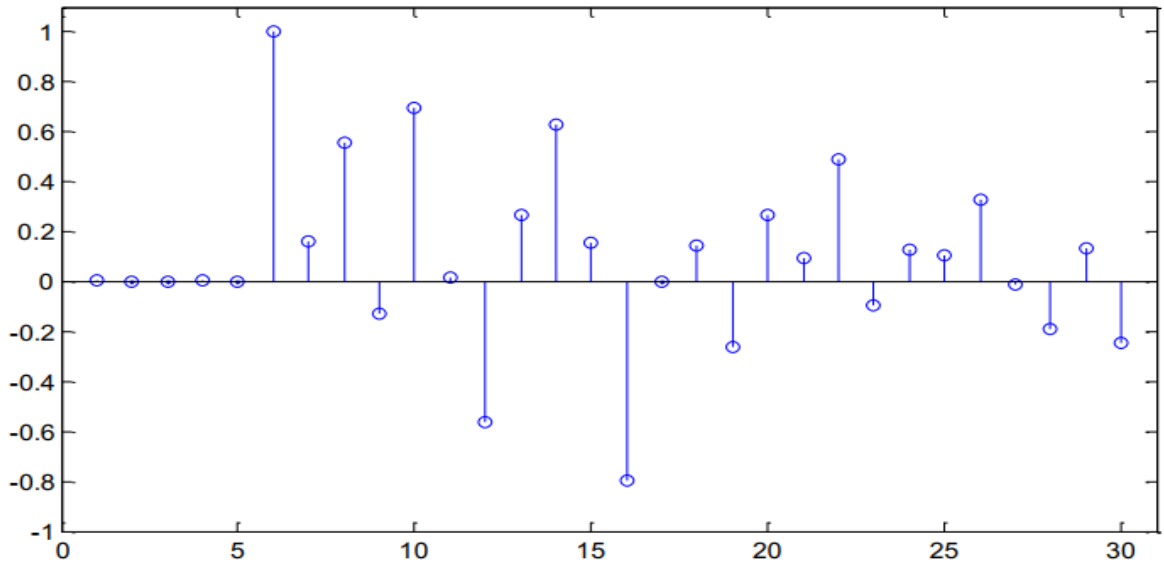


Figure1. 2: réponse impulsionnelle dispersive.

### 1.3 Réponse impulsionnelle parcimonieuse

La définition de la réponse impulsionnelle parcimonieuse est très importante dans notre étude car elle signifie que cette réponse contient un pourcentage très important des coefficients nuls, et aussi si la réponse impulsionnelle est parcimonieuse c'est-à-dire qu'une portion de son énergie est concentrée dans une durée très petite de la RI.

La réponse impulsionnelle devient comme le montre la figure (1.3), avec peu de coefficients actifs, cette réponse est dite : réponse impulsionnelle parcimonieuse (RIP).

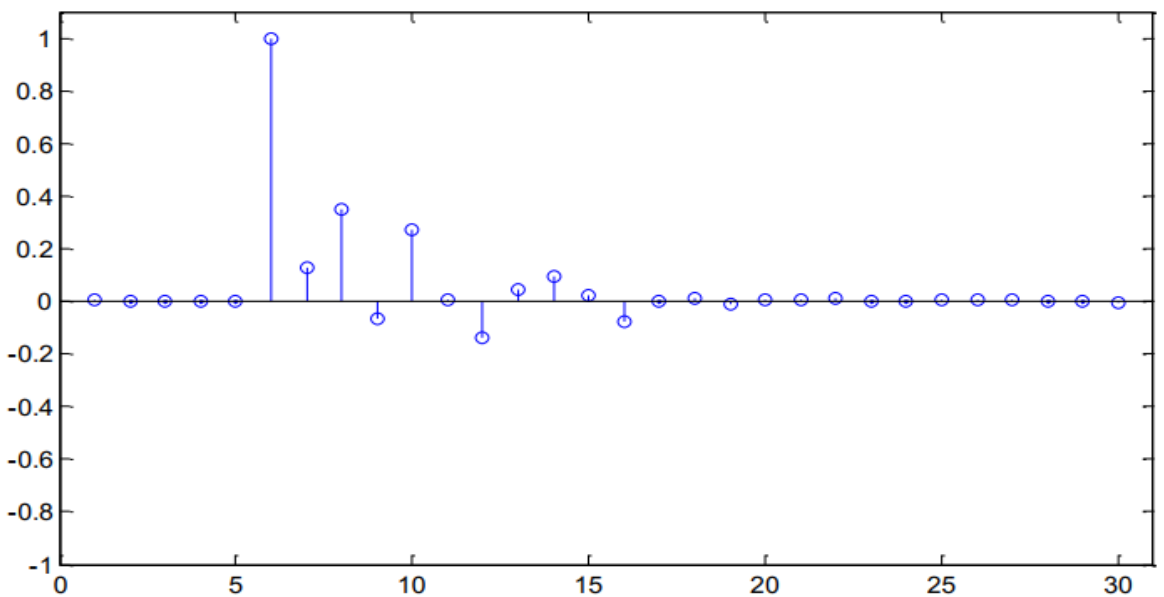


Figure1. 3: réponse impulsionnelle parcimonieuse.

## 1.4 L'écho acoustique

L'écho est le son de votre propre voix qui résonne pendant que vous parlez, l'écho peut être extrêmement perturbant dans une conversation.

L'écho acoustique provient d'une boucle audio qui s'établit lorsque 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, les personnes dans la salle distante réentendent leurs propres voix.

L'écho acoustique peut être intensifié lorsque des microphones sensibles sont utilisés, ainsi que lorsque le volume du microphone et / ou du haut-parleur est élevé, ainsi que lorsque le microphone et le ou les haut-parleurs sont positionnés de sorte que le microphone est près d'un ou de plusieurs orateurs. Cet écho est aggravé par l'écho acoustique réfléchissant réfléchi par les murs et / ou les objets, la figure (1.4) résume le problème d'écho acoustique.

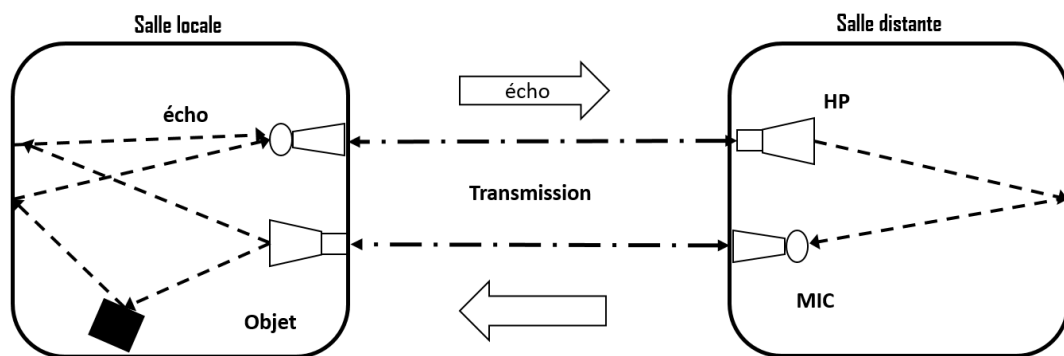


Figure1. 4: L'écho acoustique entre deux salles.

## 1.5 L'identification adaptative de RI

Le problème majeur dans l'annulation d'écho acoustique est l'identification des réponses impulsionnelles variables, beaucoup d'applications rencontrent ce genre de problèmes qui utilisent des filtres à réponse impulsionnelle finie variable aux cours de temps, donc les

filtres classiques fixes ne peuvent pas atteindre l'objectif souhaité, alors il faut un filtre intelligent, ce type de filtre est dit : filtre adaptatif. La différence la plus importante avec les filtres classiques, c'est que le filtre adaptatif peut modifier sa réponse impulsionnelle pour répondre à des critères de performance désirés.

Les filtres adaptatifs ont des applications dans différents domaines avec des objectifs spécifiques [1], parmi ces applications on cite :

- 1- L'identification de réponse impulsionnelle (modélisation adaptative).
- 2- Egalisation et déconvolution (modélisation adaptative inverse).
- 3- Prédiction linéaire adaptative.
- 4- Annulation d'interférence adaptative.

Dans notre mémoire l'utilisation des filtres adaptatifs est pour l'identification de réponse impulsionnelle acoustique qui nous permet d'annuler l'écho acoustique.

Le filtre adaptatif estime les paramètres d'un système inconnu voir figure (1.5), le signal d'entrée  $x_n$  est injecté à l'entrée de filtre adaptatif  $w$  et le système inconnu, la sortie de filtre  $\hat{y}_n$  est comparée avec le signal  $d_n$  la sortie de système inconnu plus le bruit, en faisant la soustraction entre ces deux signaux ( $\hat{y}_n$  et  $d_n$ ), le résultat est un signal d'erreur  $\tilde{\epsilon}_n$  qui est réinjecté à l'entrée de l'algorithme adaptatif comme une contre réaction, puis le filtre ajuste ses coefficients jusqu'à l'obtention d'une valeur d'erreur  $\tilde{\epsilon}_n$  qui minimise un critère donné, donc le système inconnu est modélisé et identifié par ce mécanisme.

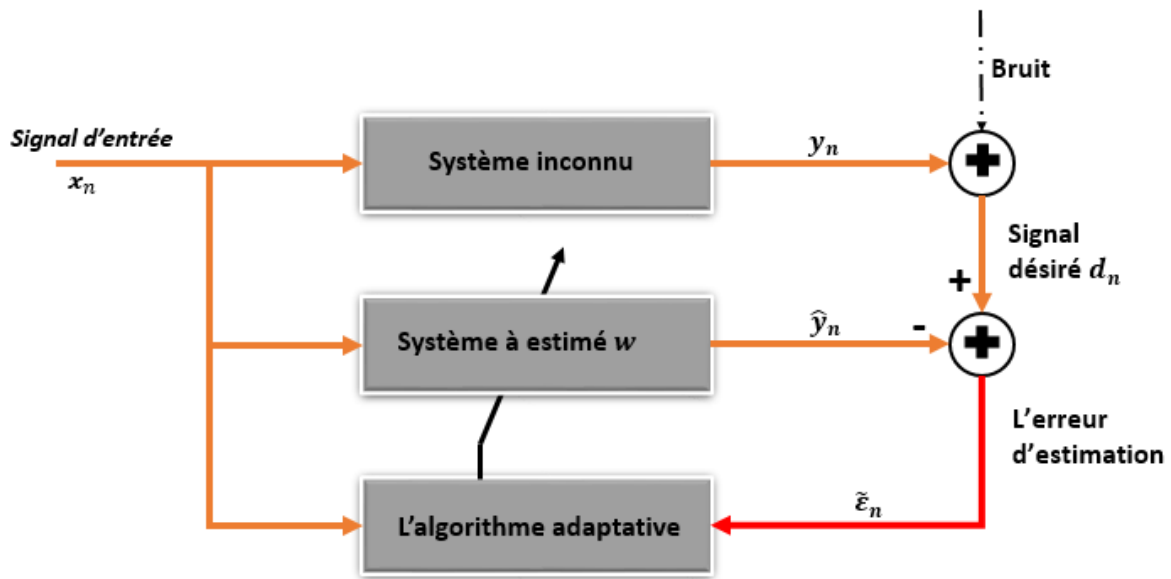


Figure1. 5: Le concept d'indentification adaptative du système.

## 1.6 Annulation d'écho acoustique

L'annulation d'écho acoustique basée sur une méthode qui traite le signal capté par le microphone, c'est-à-dire d'utiliser un système qui devrait idéalement supprimer tout ce qui provient du microphone de prise de son sauf le signal double parole.

Le microphone dans la figure 1.6 capte à la fois la parole de 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), puis il retranche cette estimation du signal local de parole. En effet, l'annulation de cet écho acoustique est basée sur l'estimation du canal acoustique entre le haut-parleur et le microphone par des techniques d'identification adaptative de la réponse impulsionnelle.

L'annulation d'écho acoustique consiste à utiliser un filtre qui génère une estimée de l'écho réel, la différence entre les deux est l'écho résiduel (erreur d'estimation), le but est de minimiser l'erreur d'estimation pour réduire au mieux l'écho, alors que la technique d'annulation d'écho devient un problème de minimisation ceci nous permet d'utiliser des filtres optimums décrits dans la section suivante.

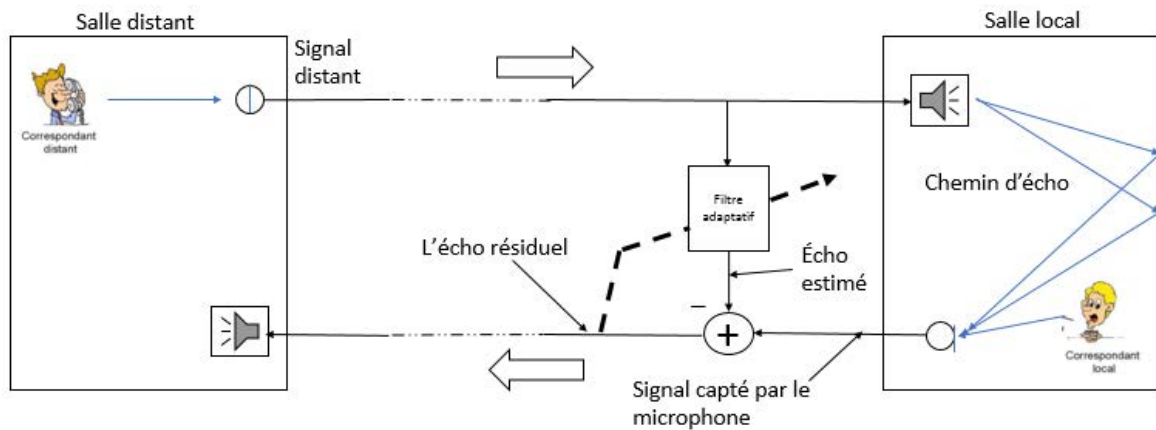


Figure1. 6: Principe d'annulation d'écho acoustique.

## 1.7 Définitions et formules

Dans la théorie des filtres adaptatifs plusieurs types sont développés, le choix de l'un entre eux est basé sur les critères suivants :

- a. **Vitesse de convergence** : c'est le nombre d'itération pour trouver le filtre optimum, plus le nombre d'itération est petite plus la performance en vitesse de convergence est meilleure.
- b. **Erreur en régime permanent** : c'est la valeur la plus petite de l'erreur quadratique moyenne.
- c. **La capacité de poursuite** : c'est la capacité de poursuivre les variations de la réponse impulsionnelle acoustique ; c'est à dire les non stationnarités du canal acoustique.
- d. **Complexité des calculs** : c'est le nombre d'opérations arithmétiques, le but c'est de réduire la complexité de calcul en gardant la performance optimale de filtre.

Les filtres adaptatifs qui sont utilisés pour estimer le signal désiré sont classés sous deux grandes catégories :

- i. Méthode de gradient stochastique.
- ii. Estimation des moindres carrées.

### 1.7.1 Le filtre optimum

En utilisant la théorie d'optimisation [2] le filtre optimal tente de trouver la solution optimale, généralement parlant ce critère est de minimiser le moyen carré d'erreur qui est la différence entre le signal de sortie de filtre et le signal désiré.

Le filtre optimal est déjà développé par Norbert Wiener, ce filtre est connu sous le nom de filtre de Wiener dont la sortie est donnée par :

$$\hat{y}_n = \mathbf{w}_{n-1}^T \mathbf{x}_n \quad (1.1)$$

Où

$$\mathbf{x}_{L,n} = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}] \quad (1.1.a)$$

c'est le vecteur du signal d'entrée  $x_n$ , et

$$\mathbf{w}_{L,n} = [w_{1,n} \quad w_{2,n} \quad \cdots \quad w_{L,n}] \quad (1.1.b)$$

est le vecteur des coefficients de filtre, alors le signal d'erreur a priori est calculée par :

$$\bar{e}_n = d_n - \hat{y}_n \quad (1.2)$$

Le but de l'algorithme est d'estimer le système le plus proche du système réel, cette estimation est réalisée en minimisant la fonction de coût  $J_n(\mathbf{w})$  qui dépend de l'erreur  $e_n$ , Les algorithmes que nous allons décrire dans la prochaine section se basent sur cette méthode de trouver le filtre optimal qui minimise la fonction coût suivante :

$$J_n(\mathbf{w}) = E\{(d_n - \mathbf{w}_n^T \mathbf{x}_n)^2\} \quad (1.3)$$

L'objectif ultime est de trouver l'erreur le plus petit possible pour le filtre optimal qui aura des coefficients plus proches que le signal désiré, maintenant c'est un problème d'optimisation, alors le filtre optimal est donné par :

$$\mathbf{w}_{opt} = \arg \min_{\mathbf{w}} J_{EMQ}(\mathbf{w}) \quad (1.4)$$

L'équation (1.4) a une forme quadratique alors sa solution optimale est quand la dérivée de  $J_n(\mathbf{w})$  est nulle :

$$\nabla_{\mathbf{w}} J_{EMQ}(\mathbf{w}) = \frac{\partial J_{EMQ}}{\partial \mathbf{w}} = 0 \quad (1.5)$$

$$2\mathbf{R}_{xx} \mathbf{w}_n - 2\mathbf{p}_{xd} = 0 \quad (1.6)$$

Sachant que :

$$\mathbf{R}_{xx} = E\{\mathbf{x}_n \mathbf{x}_n^T\}$$

Est la matrice d'autocorrélation du vecteur  $\mathbf{x}_n$ , et :

$$\mathbf{p}_{xd} = E\{\mathbf{x}_n d_n\}$$

Est le vecteur d'inter-corrélation du signal désiré et le signal d'entrée. Si la matrice d'autocorrélation est inversible alors le filtre optimal est donné par l'expression suivante :

$$\mathbf{w}_{opt} = \mathbf{R}_{xx}^{-1} \mathbf{p}_{xd} \quad (1.7)$$

Alors pour atteindre cet objective il faut connaitre les propriétés statistiques de  $\mathbf{R}_{xx}$  et de  $\mathbf{p}_{xd}$ .

## 1.7.2 Algorithme basé sur le gradient stochastique

### a Algorithme de moindre carrée moyenne (LMS)

L'algorithme LMS est un algorithme le plus simple de la famille de gradient stochastique, la procédure itérative de cet algorithme a pour but de minimiser le gradient d'erreur quadratique moyenne EQM par une procédure d'adaptation dans la direction opposée du gradient de EQM :

$$\mathbf{w}_n = \mathbf{w}_{n-1} - \mu |\nabla_w E\{\bar{\varepsilon}_n^2\}| \quad (1.8)$$

Où  $\mu$  est le pas d'adaptation qui contrôle la convergence de l'algorithme.

L'équation 1.8 nous donne :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2 \mu E\{\bar{\varepsilon}_n \mathbf{x}_n\} \quad (1.9)$$

Alors l'équation finale de mis à jour des coefficients de filtre est comme suite :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu \bar{\varepsilon}_n \mathbf{x}_n \quad (1.10)$$

La complexité de l'algorithme LMS est 2L multiplication pour chaque itération, la condition de convergence de cet algorithme est donnée par un pas d'adaptation  $\mu$  qui satisfait la condition suivante [3] :

$$0 < \mu_{LMS} < \frac{2}{\text{trace}(\mathbf{R}_{xx})} = \frac{1}{L\sigma_x^2} \quad (1.11)$$

Où  $\text{trace}(\cdot)$  est la trace d'une matrice, l'algorithme de LMS résumé dans la table 1.1 :

<p><b>Initialisation :</b> <math>L = \text{taille de filtre}</math></p>
---

$$0 < \mu \leq \frac{1}{L\sigma_x^2}$$

$$\mathbf{w}_0 = \mathbf{0}_{L \times 1}$$

**Partie filtrage :**

**For n = 1, 2, ... (iterations)**

$$\mathbf{x}_n = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T$$

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

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu \bar{\varepsilon}_n \mathbf{x}_n$$

**End For**

Table 1. 1: L'algorithme LMS.

***b Algorithme de LMS normalisé (NLMS)***

Une variante de l'algorithme LMS gradient stochastique est l'algorithme LMS normalisé NLMS où le gain d'adaptation est normalisé par l'énergie du signal d'entrée  $x_n$  :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\bar{\varepsilon}_n \mathbf{x}_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}} \quad (1.12)$$

Où  $\delta_{NLMS}$  est une constante utilisée pour éviter la division par des petites valeurs ou des zéros, la convergence de cet algorithme est garantie par la condition suivante :

$$0 < \mu < 2$$

L'intérêt de l'algorithme NLMS par rapport au LMS est que le choix du pas d'adaptation de l'algorithme NLMS est indépendant de la variance de signal d'entrée, cet algorithme est indiqué dans la table 1.2 :

**Initialisation :**  $L = \text{taille de filtre}$

$$0 < \mu \leq 2, \quad \delta_{NLMS} = \frac{L}{100} \cdot \sigma_x^2$$

$$\mathbf{w}_0 = \mathbf{0}_{L \times 1}$$



### Partie filtrage :

**For n = 1, 2, ... (iterations)**

$$\mathbf{x}_n = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T$$

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

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\bar{\varepsilon}_n \mathbf{x}_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}}$$

**End For**

Table 1. 2: L'algorithme NLMS.

### 1.7.3 Estimation des moindres carrées

Dans cette catégorie, nous utilisons une somme  $\sum(\cdot)$  déterministe au lieu de la moyenne statistique  $E\{\cdot\}$  [4] dans la fonction de coût à minimiser :

$$J_n(\mathbf{w}) = \sum_{i=1}^n |\bar{\varepsilon}_i|^2 \quad (1.13)$$

#### *a Algorithme de moindres carrées récursifs (RLS)*

L'algorithme RLS a pour but de résoudre le problème de EQM minimale récursivement pour chaque point de signal d'entrée, la fonction de coût de l'algorithme RLS est donnée par :

$$J_n(\mathbf{w}) = \sum_{i=1}^n \lambda^{n-i} (d_i - \mathbf{w}_n^T \mathbf{x}_i)^2 \quad (1.14)$$

Où  $\lambda$  est le facteur d'oubli, pour négliger les informations passées qui n'ont pas un effet sur le mis à jour des coefficients du filtre.

En minimisant l'équation ci-dessus, on obtient le vecteur  $\mathbf{w}_n$  qui minimise la fonction de coût d'erreur de RLS comme suite :

$$\frac{\partial J_{EMQ}}{\partial \mathbf{w}} = -2 \sum_{i=1}^n \lambda^{n-i} e_i^2 \mathbf{x}_i [d_i - \mathbf{x}_i^T \mathbf{w}_n] \quad (1.15)$$

$$\frac{\partial J_{EMQ}}{\partial \mathbf{w}} = -2 [\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i d_i - (\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T) \mathbf{w}_n] \quad (1.16)$$

En égalant l'équation 1.16 au zéro on obtient :

$$[\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T] \mathbf{w}_n = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i d_i \quad (1.17)$$

On note le terme  $\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^T$  par  $\mathbf{R}_n$  la matrice d'autocorrélation court-terme, et le terme  $\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_i d_i$  par  $\mathbf{p}_n$  le vecteur d'inter-corrélation court-terme, alors l'équation 1.17 devient :

$$\mathbf{R}_n \mathbf{w}_n = \mathbf{p}_n \quad (1.18.a)$$

et par conséquence on a :

$$\mathbf{w}_n = \mathbf{R}_n^{-1} \mathbf{p}_n \quad (1.18.b)$$

Alors la mise à jour des coefficients de filtre nécessite l'inverse de matrice d'autocorrélation de signal d'entrée. La complexité des calculs est de l'ordre  $L^3$  opérations arithmétiques par itération.

L'équation récursive de la matrice d'autocorrélation se calcule par :

$$\mathbf{R}_{L,n} = \lambda \sum_{i=1}^n \lambda^{n-1-i} \mathbf{x}_i \mathbf{x}_i^T + \mathbf{x}_n \mathbf{x}_n^T \quad (1.19)$$

$$= \lambda \mathbf{R}_{n-1} + \mathbf{x}_n \mathbf{x}_n^T \quad (1.20)$$

Et pour le vecteur d'inter-corrélation :

$$\mathbf{p}_n = \lambda \mathbf{p}_{n-1} + \mathbf{x}_n d_n^T \quad (1.21)$$

En utilisant l'équation (1.20) et (1.21) dans (1.18.a) :

$$\lambda \mathbf{R}_{n-1} \mathbf{w}_{n-1} = \lambda \mathbf{p}_{n-1} \quad (1.22)$$

La mise à jour des coefficients de filtre récursive est donnée par :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \bar{\varepsilon}_n \mathbf{k}_n \quad (1.23)$$

$$\mathbf{k}_n = \mathbf{R}_n^{-1} \mathbf{x}_n \quad (1.24)$$

Où  $\mathbf{k}_n$  est le gain de Kalman.

En appliquant à la matrice  $\mathbf{R}_n$  le lemme d'inversion matricielle suivant [2] :

$$(\mathbf{A} + \mathbf{U}\mathbf{V}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{U}\mathbf{V}^T\mathbf{A}^{-1}}{1 + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}} \quad (1.25)$$

Avec

$$A = \lambda \mathbf{R}_{n-1} \quad ; \quad U = V = \mathbf{x}_n$$

Il vient

$$\mathbf{R}_n^{-1} = \lambda^{-1} \mathbf{R}_{n-1}^{-1} - \frac{\mathbf{R}_{n-1}^{-1} \mathbf{x}_n \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1}}{\lambda + \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1} \mathbf{x}_n} \quad (1.26)$$

La complexité de cet algorithme est élevée ( $L^2$ ) par rapport au type de gradient stochastique, la description de l'algorithme est ci-dessous dans la table 1.3:

<p><b>Initialisation :</b>     <math>L = \text{taille de filtre}</math></p> <p><math>\mathbf{w}_0 = \mathbf{k}_0 = \mathbf{0}_{L \times 1}</math> ; <math>\mathbf{R}_0^{-1} = \delta^{-1} \mathbf{I}_L</math> ; <math>\lambda &lt; 1</math></p>
<p><b><u>Partie filtrage :</u></b></p> <p><b>For n = 1, 2, ... (iterations)</b></p> <p><math>\mathbf{x}_n = [x_n \quad x_{n-1} \quad \dots \quad x_{n-L+1}]^T</math></p> <p><math>\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}</math></p> <p><math>\bar{\varepsilon}_n = d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}</math></p> <p><math>\mathbf{w}_n = \mathbf{w}_{n-1} + \bar{\varepsilon}_n \mathbf{k}_n</math></p> <p><math>\mathbf{R}_n^{-1} = \lambda^{-1} [\mathbf{R}_{n-1}^{-1} - \mathbf{k}_n \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1}]</math></p>

Table 1. 3: L'algorithme RLS.

## 1.8 Conclusion

Dans ce chapitre, nous avons présenté quelques notions sur la réponse impulsionnelle acoustique et réponse impulsionnelle parcimonieuse ainsi que le concept d'annulation d'écho acoustique, ensuite nous avons décrits quelques définitions sur les filtres adaptatifs que nous allons les utilisées dans les prochains chapitres.

# Chapitre 2 Algorithmes d'identifications parcimonieux

---

## 2.1 Introduction

Ces dernières années, la notion de parcimonie a fait l'objet d'un grand intérêt de part de la communauté du traitement du signal dans les conférences internationales. Incontestablement, la parcimonie est devenue une notion incontournable en traitement du signal. Aujourd'hui, on ne compte plus le nombre de publications citant le terme parcimonie.

Dans ce chapitre, nous allons discuter à propos d'une famille d'algorithmes de type gradient stochastique NLMS pour l'identification des réponses impulsionnelles parcimonieuses ; notre intérêt c'est porté sur la catégorie d'algorithmes zero attracting NLMS et ses variantes, puis nous proposons une extension de ces principes de parcimonie au cas d'algorithmes fast NLMS.

Par la suite nous faisons intervenir la notion de parcimonie sur les filtres de moindres carrés, cela nous permet de définir la méthode de EM (Expectation-Maximisation) pour l'algorithme RLS (EM-RLS).

## 2.2 Les algorithmes adaptatifs pour les RI parcimonieuses

Une méthode courante pour exploiter la parcimonie de la réponse impulsionnelle consiste à modifier la fonction de coût en ajoutant une pénalité de norme  $\ell_1$  [6] ou une pénalité de norme  $\ell_0$  au carré de l'erreur instantanée [7], dans la suite nous allons présenter les algorithmes qui utilisent cette méthode de pénalité.

### 2.2.1 Algorithme zero attracting NLMS (ZA-NLMS)

L'utilisation de la méthode LASSO [8] permet d'introduire deux contraintes de parcimonie différentes ( $\ell_1$ -norme et la fonction de pénalité de la somme de log) dans la fonction de

coût quadratique convexe de l'algorithme NLMS, nous donne deux nouveaux algorithmes qui s'appellent ZA-NLMS (Zero Attracting LMS) et RZA-NLMS (reweighted zero attracting NLMS) où ces algorithmes utilisent la règle de mise à jour suivante :

$$\begin{pmatrix} \text{nouvel} \\ \text{parametre} \\ \text{estimé} \end{pmatrix} = \begin{pmatrix} \text{ancienne} \\ \text{parametre} \\ \text{estimé} \end{pmatrix} + (\text{pas d'adaptation}) \begin{pmatrix} \text{nouvel} \\ \text{inforamtion} \end{pmatrix} + \begin{pmatrix} \text{Terme} \\ \text{d'attraction} \\ \text{zéro} \end{pmatrix}$$

Le terme zéro -attraction (ZA) (ou zéro attracteur) est une fonction de régularisation normée qui applique une attraction à zéro sur les petits paramètres [9].

Dans le ZA-NLMS, une nouvelle fonction de coût est définie en combinant l'erreur quadratique instantanée avec la pénalité de norme  $\ell_1$  du vecteur des coefficients comme suit :

$$J_{n,ZA-NLMS}(w) = \frac{1}{2} \bar{\varepsilon}_n^2 + \lambda_{ZA} \|\mathbf{w}_n\|_1 \quad (2.1)$$

Ce qui signifie que la fonction de coût parcimonieuse  $J_{ZA-NLMS}$  combine l'erreur instantanée  $\bar{\varepsilon}_n$  avec un terme de pénalité de faible densité ( $\lambda_{ZA} \|\mathbf{w}_n\|_1$ ).

Avec  $\lambda_{ZA}$  est un paramètre de régularisation pour équilibrer l'erreur d'estimation et la pénalité parcimonieuse de  $\mathbf{w}_n$ .

L'équation de mise à jour correspondante [10] de ZA-NLMS :

$$\begin{aligned} \mathbf{w}_n &= \mathbf{w}_{n-1} - \mu \left[ \frac{\partial J_{n,ZA-NLMS}(w)}{\partial \mathbf{w}_{n-1}} \right] \\ &= \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}} - \rho_{ZA} \cdot \text{sgn}(\mathbf{w}_{n-1}) \end{aligned} \quad (2.2)$$

Où  $\rho_{ZA}$  est le paramètre qui contrôle le taux de convergence de l'algorithme vers la solution optimale  $\rho_{ZA} = \mu \lambda_{ZA}$  [11]. La fonction  $\text{sgn}(\cdot)$  est une fonction de signe définie comme :

$$\text{sgn}(w) = \begin{cases} 1, & w > 0 \\ 0, & w = 0 \\ -1, & w < 0 \end{cases}$$

En observant l'équation de mise à jour (2.2), son deuxième terme attire les coefficients de filtre de petite valeur à zéro. L'effet de cette opération est d'augmenter la vitesse de convergence [12], l'algorithme de ZA-NLMS est présenté dans la table 2.1.

<p><b>Initialisation :</b>     <math>L = \text{taille de filtre}</math></p> <p><math>0 &lt; \mu \leq 2, \delta_{NLMS} = \frac{L}{100} \cdot \sigma_x^2, \rho_{ZA} = 1 \times 10^{-5}</math></p> <p><math>\mathbf{w}_0 = \mathbf{0}_{L \times 1}</math></p>
<p><b>Partie filtrage :</b></p> <p><b>For n = 1, 2, ... (iterations)</b></p> <p><math>\mathbf{x}_n = [x_n \ x_{n-1} \ \dots \ x_{n-L+1}]^T</math></p> <p><math>\bar{\varepsilon}_n = d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}</math></p> <p><math>\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}} - \rho_{ZA} \cdot \text{sgn}(\mathbf{w}_{n-1})</math></p> <p><b>End For</b></p>

Table 2. 1: L'algorithme ZA-NLMS.

## 2.2.2 Algorithme Reweighted Zero Attracting NLMS (RZA-NLMS)

Dans cet algorithme en utilisant la pénalité de somme-log car elle a un comportement similaire à la norme  $\ell_0$  que la norme  $\ell_1$  dans la fonction de coût de l'algorithme NLMS, un algorithme robuste a été dérivé pour l'identification des RI parcimonieuse, dans [11] ils proposent une approche heuristique pour renforcer l'attracteur zéro appeler reweighted zero-attracting NLMS (RZA-NLMS) dont la fonction de coût s'écrit comme :

$$J_{n,RZA-NLMS}(\mathbf{w}) = \frac{1}{2} \bar{\varepsilon}_n^2 - \lambda_{RZA} \sum_{n=1}^L \log(1 + \varepsilon_{RZA} |\mathbf{w}_n|) \quad (2.3)$$

Avec  $\lambda_{RZA}$  c'est un paramètre de régulation sachant que  $\varepsilon_{RZA} > 0$ , où l'équation de mise à jour [10,13] s'écrit sous forme :

$$\mathbf{w}_n = r \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}} - \rho_{RZA} \frac{\text{sgn}(\mathbf{w}_{n-1})}{1 + \varepsilon_{RZA} |\mathbf{w}_{n-1}|} \quad (2.4)$$

Où  $\rho_{RZA} = \mu \lambda_{RZA} \varepsilon_{RZA}$  avec  $|\mathbf{w}_{n-1}|$  c'est un vecteur de  $L \times 1$  éléments, le terme  $\frac{\text{sgn}(\mathbf{w}_{n-1})}{1 + \varepsilon_{RZA} |\mathbf{w}_{n-1}|}$  est obtenu par division d'élément par élément.

Dans l'équation (2.4), lorsque  $\varepsilon_{RZA}$  est très petit, le terme  $\varepsilon_{RZA} |\mathbf{w}_{n-1}|$  tend vers zéro, donc l'équation de mise à jour de RZA-NLMS et l'algorithme de ZA-NLMS deviennent similaire [14], l'algorithme de RZA-NLMS est aussi décrit dans la table suivante :

<p><b>Initialisation :</b> <math>L = \text{taille de filtre}</math></p> <p><math>0 &lt; \mu \leq 2, \delta_{NLMS} = \frac{L}{100} \cdot \sigma_x^2, \rho_{ZAN} = 1 \times 10^{-5}; \varepsilon_{RZA} = 50</math></p> <p><math>\mathbf{w}_0 = \mathbf{0}_{L \times 1}</math></p>
<p><b>Partie filtrage :</b></p> <p><b>For n = 1, 2, ... (iterations)</b></p> <p><math>\mathbf{x}_n = [x_n \quad x_{n-1} \quad \dots \quad x_{n-L+1}]^T</math></p> <p><math>\bar{\varepsilon}_n = d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}</math></p> <p><math>\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n e_n}{\mathbf{x}_n^T \mathbf{x}_n + \delta_{NLMS}} - \rho_{RZA} \frac{\text{sgn}(\mathbf{w}_{n-1})}{1 + \varepsilon_{RZA}  \mathbf{w}_{n-1} }</math></p> <p><b>End For</b></p>

Table 2. 2: L'algorithme RZA-NLMS.

### 2.2.3 L'effet de $\rho_{(R)ZA}$ ( $\rho_{ZA}$ ou $\rho_{RZA}$ )

Dans les filtres adaptatifs le paramètre de régularisation joue un rôle très important pour la convergence de cet algorithme, dans l'équation (2.2) et (2.4), une grande valeur de  $\rho_{(R)ZA}$  nous donne une vitesse de convergence très élevée et l'intensité d'attraction augmente proportionnellement avec le paramètre  $\rho_{(R)ZA}$  ce signifie que l'erreur en régime permanent diminue lorsque  $\rho_{(R)ZA}$  augmente.

Le paramètre  $\rho_{(R)ZA}$  est donc déterminé par le compromis entre la vitesse d'adaptation et l'erreur en régime permanent dans une application particulière [14].

## 2.3 Algorithme des moindres carrés rapide

Dans les applications d'annulation d'écho acoustique qui ont été expliquées dans le chapitre 1, on estime le filtre transversal inconnu à partir du signal d'entrée  $x_{L,n}$  et le signal désiré  $d_n$ , ce filtre nécessite entre 200 et 2000 coefficients d'après [15], ce qui conduit à un coût de calculs très élevé pour l'identification en temps réel. Pratiquement, il est utile de réduire la complexité des calculs pour l'implémentation en temps réel, dans [16] [17], les auteurs ont déjà réduit la complexité des calculs de l'algorithme des moindres carrés rapide (Fast Transversal Filter FTF) utilisant seulement un prédicteur direct de taille réduite pour calculer le gain d'adaptation avec  $2L+5P$  multiplications.

Dans cette section nous utilisons le même algorithme FTF où le gain d'adaptation est obtenu à partir d'un prédicteur du 1<sup>er</sup> ordre. ; C'est l'algorithme type NLMS à convergence rapide (FNLMS).

### 2.3.1 Algorithme Fast convergence NLMS (FNLMS)

Le problème des filtres adaptatifs est schématisé dans la figure (1.5), où  $x_n$  est le signal d'entrée et  $d_n$  est le signal désiré.

L'erreur de filtre adaptatif est donnée par :

$$\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n \quad (2.5)$$

La mise à jour de filtre  $w_{L,n}$  est définie par :

$$\mathbf{w}_n = \mathbf{w}_{n-1} - \mathbf{c}_n \bar{\varepsilon}_n \quad (2.6)$$

Avec  $\mathbf{c}_n$  est le vecteur de gain d'adaptation. Dans l'algorithme FTF [18], le gain d'adaptation pour (2.6) est :

$$\mathbf{c}_n = \gamma_n \tilde{\mathbf{c}}_n \quad (2.7)$$

Où  $\tilde{\mathbf{c}}_n$  est le gain de kalman dual, et  $\gamma_n$  est la variable de vraisemblance, ces deux variables sont obtenues par la propagation de deux vecteurs  $\mathbf{a}_n$  et  $\mathbf{b}_n$ , qui sont la prédiction aller et retour respectivement [16,18] :



$$\begin{bmatrix} \tilde{\mathbf{c}}_n \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{c}}_n \end{bmatrix} - \frac{\bar{e}_{L,n}}{\lambda \alpha_{n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{n-1} \end{bmatrix} + \frac{\bar{r}_n}{\lambda \beta_{n-1}} \begin{bmatrix} -\mathbf{b}_{n-1} \\ 1 \end{bmatrix} \quad (2.8)$$

Où  $\bar{e}_{L,n}$  et  $\bar{r}_{L,n}$  sont respectivement les erreurs de prédiction aller et retour, et  $\alpha_{L,n}$  et  $\beta_{L,n}$  sont les variances des erreurs prédiction respectivement, la variable  $\gamma_{L,n}$  est donnée par :

$$\gamma_n = \frac{1}{1 - \tilde{\mathbf{c}}_n^T \mathbf{x}_n} \quad (2.9)$$

L'algorithme de FTF est numériquement instable quand il est implémenté dans une arithmétique de précision finie, il y a plusieurs solutions pour la stabilisation avec des signaux stationnaires sont proposés dans la littérature, le taux de complexité de FTF stabilisé (SFTF) est 8L opérations par itération [16].

Dans [16] [17], les auteurs proposent de remplacer (2.8) par la relation simplifiée suivante :

$$\begin{bmatrix} \tilde{\mathbf{c}}_n \\ \mathbf{c}_n \end{bmatrix} = \begin{bmatrix} \frac{-e_n}{\lambda \alpha_{n-1} + c_0} \\ \tilde{\mathbf{c}}_{n-1} \end{bmatrix} \quad (2.10)$$

On utilise seulement l'erreur de prédiction de signal d'entrée, notée  $e_n$  au lieu  $\bar{e}_{L,n}$  pour évaluer le gain de kalman dual où  $c_0$  est une petite constante positive pour éviter la division par des petites valeurs en absence du signal d'entrée, et la variance d'erreur de prédiction aller est évaluée par :

$$\alpha_n = \lambda \alpha_{n-1} + e_n^2 \quad (2.11)$$

L'erreur de prédiction dans (2.11) peut être calculer indépendamment du gain de Kalman par n'importe quel autre algorithme de prédiction adaptatif. L'opération de prédiction qui donne  $e_n$  à partir de signal d'entrée  $x_n$  est connue aussi par le filtre de décorrélation [19], cette technique est utilisée pour améliorer la rapidité de convergence de l'algorithme type NLMS. Dans l'algorithme de FNLMS, l'erreur de prédiction est calculée par le modèle de premier ordre de prédiction :

$$e_n = x_n - ax_{n-1} \quad (2.12)$$

Avec  $a$  est le paramètre de prédiction, ce paramètre est fixé si nous possédons une information a priori sur le signal d'entrée  $x_n$ , pour le bruit blanc  $a = 0$  et pour le bruit USASI le premier coefficient de prédiction  $a = 0.925$ , généralement parlant le signal d'entrée

varie dans le temps donc ses statistiques sont inconnues, alors le coefficient  $a$  doit être estimé à partir du signal d'entrée, on considère ce coefficient qui minimise la fonction de coût  $E\{e_n^2\}$  et en l'égalant avec le zéro la dérivé de cette fonction de coût,  $a$  est donné par :

$$a = \frac{E\{x_n x_{n-1}\}}{E\{x_{n-1}^2\}} = \frac{r_1}{r_0} \quad (2.13)$$

Où  $r_0$  et  $r_1$  sont respectivement la puissance de signal d'entrée et le premier coefficient de la fonction d'autocorrélation pour un signal d'entrée stationnaire, alors le coefficient de prédiction est décrit par :

$$a_n = \frac{r_{1,n}}{r_{0,n} + c_0} \quad (2.14)$$

Sachant que  $r_{1,n}$  et  $r_{0,n}$  sont estimés respectivement par :

$$r_{1,n} = \lambda_a r_{1,n-1} + x_n x_{n-1} \quad (2.15)$$

$$r_{0,n} = \lambda_a r_{0,n-1} + x_n^2 \quad (2.16)$$

Avec  $\lambda_a$  est le facteur d'oubli exponentiel, et  $c_0$  est une petite constante positive.

Plusieurs méthodes pour calculer  $\gamma_{L,n}$  sont possibles, la première méthode est d'utiliser la fonction (2.9), la complexité de calcul dans cette méthode est à l'ordre  $3L$  multiplications par itération, la deuxième méthode exploite certaines propriétés de décalage après la première composant de  $\tilde{c}_n^1$  du gain de kalman dual pour calculer  $\gamma_n$  avec peu d'opérations par itération :

$$\gamma_n = \frac{1}{1 - \sum_{i=1}^L \tilde{c}_{n-i+1}^1 x_{n-i+1}} \quad (2.17)$$

Cette équation peut s'écrire comme suite :

$$\gamma_n = \frac{1}{1 - \sum_{i=1}^L v_{n-i+1}} \quad (2.18)$$

Ou  $v_n = \tilde{c}_n^1 x_n$  est le signal utilisé dans le processus de décalage la complexité des calculs est proportionnelle à  $2L$  multiplications.

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

$$\mathbf{x}_{L+1,n}^T = [\mathbf{x}_n^T \quad x_{n-L}] \quad (2.19.a)$$

Et

$$\mathbf{x}_{L+1,n}^T = [x_n \quad \mathbf{x}_{n-1}^T] \quad (2.19.b)$$

Par la multiplication de (2.19.a) par le premier membre de l'équation (2.10), et (2.19.b) par le deuxième membre de l'équation (2.10), nous donne l'égalité suivante :

$$\mathbf{x}_n^T \tilde{\mathbf{c}}_n + c_n x_{n-L} = \mathbf{x}_{n-1}^T \tilde{\mathbf{c}}_{n-1} - \frac{x_n e_n}{\lambda \alpha_{L,n-1} + c_0} \quad (2.20)$$

En remplaçant  $\mathbf{x}_n^T \tilde{\mathbf{c}}_n$  par  $1 - \gamma_n$  dans (2.23), on obtient l'équation de mise à jour récursive pour  $\gamma_{L,n}$  :

$$\gamma_n^{-1} = \gamma_{n-1}^{-1} + \mathbf{c}_n x_{n-L} + \frac{x_n e_n}{\lambda \alpha_{n-1} + c_0} \quad (2.21)$$

L'équation (2.21) peut s'écrire par :

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \gamma_{n-1} \delta_n} \quad (2.22)$$

Où

$$\delta_n = \mathbf{c}_n x_{n-L} + \frac{x_n e_n}{\lambda \alpha_{n-1} + c_0} \quad (2.23)$$

Partie filtrage :

$$\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n \quad (2.24)$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{c}}_n \quad (2.25)$$

La complexité des calculs totale pour (2.25) est 2L multiplications, l'algorithme de FNLMS est indiqué ci-dessous (table 2.4) :

**Initialisation :**  $L = \text{taille de filtre}$

$\mathbf{w}_0 = \mathbf{c}_0 = \mathbf{0}_{L \times 1}$  ;  $\gamma_{L,0} = 1$  ;  $\alpha_0 = r_{0,0} = E_0$  ;  $r_{1,0} = 0$  ;  $c_0 = 0.01$

**For n = 1, 2, ... (iterations)**

$$\mathbf{x}_n = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T$$

**Génération de :  $e_n$**

$$r_{1,n} = \lambda_a r_{1,n-1} + x_n x_{n-1}$$

$$r_{0,n} = \lambda_a r_{0,n-1} + x_n^2$$

$$a_n = \frac{r_{1,n}}{r_{0,n} + c_0}$$

$$e_n = x_n - a_n x_{n-1}$$

$$\alpha_n = \lambda \cdot \alpha_{n-1} + e_n^2$$

**Gain d'adaptation :**

$$\begin{bmatrix} \tilde{\mathbf{c}}_n \\ c_n \end{bmatrix} = \begin{bmatrix} \frac{-e_n}{\lambda \cdot \alpha_{n-1} + c_0} \\ \tilde{\mathbf{c}}_{n-1} \end{bmatrix}$$

$$\delta_n = c_n x_{n-L} + \frac{x_n e_n}{\lambda \cdot \alpha_{n-1} + c_0}$$

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \gamma_{n-1} \delta_n}$$

**Filtrage :**

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

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

Table 2. 3: L'algorithme FNLMS.

## 2.4 Les algorithmes FNLMS parcimonieux

Dans le but d'améliorer la vitesse de convergence de l'algorithme FNLMS avec des réponses impulsionnelles parcimonieuses, nous allons appliquer la notion décrite dans 2.2 dans cet algorithme (FNLMS) pour atteindre le but cité au début.

### 2.4.1 Algorithme zero-attracting FNLMS (ZA-FNLMS)

Comme nous avons déjà parlé dans la section 2.2.1 sur l'algorithme ZA-NLMS et sa convergence par rapport au l'algorithme de NLMS, dans cette section on va essayer d'introduire la notion de pénalité sur l'algorithme de FNLMS en respectant les équations de prédictions de cet algorithme, donc le changement se fait au niveau de l'équation de coût et l'équation de mise à jour :

$$J_{n,ZA-FNLMS}(w) = \frac{1}{2} \bar{\varepsilon}_n^2 + \lambda_{ZA} \|w_n\|_1 \quad (2.26)$$

$$w_n = w_{n-1} + \mu \bar{\varepsilon}_n \gamma_n \tilde{c}_n - \rho_{ZA} \text{sign}(w_{n-1}) \quad (2.27)$$

Le détail de l'algorithme ZA-FNLMS se résume dans le tableau suivant :

<p><b>Initialisation :</b>     <math>L = \text{taille de filtre}</math></p> <p><math>w_0 = c_0 = \mathbf{0}_{L \times 1}</math> ; <math>\gamma_{L,0} = 1</math> ; <math>\alpha_0 = r_{0,0} = E_0</math> ; <math>r_{1,0} = 0</math> ; <math>\rho_{ZAF} = 1 \times 10^{-5}</math></p> <p><math>c_0 = 0.01</math></p>
<p><b>For n = 1, 2, ... (iterations)</b></p> <p><math>\mathbf{x}_n = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T</math></p> <p><b>Génération de : <math>e_n</math></b></p> <p><math>r_{1,n} = \lambda_a r_{1,n-1} + x_n x_{n-1}</math></p> <p><math>r_{0,n} = \lambda_a r_{0,n-1} + x_n^2</math></p> <p><math>a_n = \frac{r_{1,n}}{r_{0,n} + c_0}</math></p>

$e_n = x_n - a_n x_{n-1}$ $\alpha_n = \lambda \cdot \alpha_{n-1} + e_n^2$ <p><b>Gain d'adaptation :</b></p> $\begin{bmatrix} \tilde{c}_n \\ c_n \end{bmatrix} = \begin{bmatrix} \frac{-e_n}{\lambda \cdot \alpha_{n-1} + c_0} \\ \tilde{c}_{n-1} \end{bmatrix}$ $\delta_n = c_n x_{n-L} + \frac{x_n e_n}{\lambda \cdot \alpha_{n-1} + c_0}$ $\gamma_n = \frac{\gamma_{n-1}}{1 + \gamma_{n-1} \delta_n}$ <p><b>Filtrage :</b></p> $\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n$ $\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{c}}_n - \rho_{ZA} \text{sign}(\mathbf{w}_{n-1})$
--

Table 2. 4: L'algorithme ZA-FNLMS.

### 2.4.2 Algorithme Reweighted zero-attracting FNLMS (RZA-FNLMS)

De même que la section 2.2.2, en utilisant la pénalité de somme-log, l'équation de mise à jour de ZA-FNLMS devient :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{c}}_n - \rho_{RZA} \frac{\text{sgn}(\mathbf{w}_n)}{1 + \varepsilon_{RZA} |\mathbf{w}_{n-1}|} \quad (2.28)$$

Ce qui change par rapport au l'algorithme principal FNLMS, c'est la partie filtrage qui est renforcée par le terme de pénalité qui attire les petites valeurs vers zéro, pour cela le terme  $\rho_{RZA} \frac{\text{sgn}(\mathbf{w}_n)}{1 + \varepsilon_{RZA} |\mathbf{w}_{n-1}|}$  est introduit dans la partie filtrage de l'équation (2.28). Cet algorithme

résumé dans le tableau suivant :

<p><b>Initialisation :</b> <math>L = \text{taille de filtre}</math></p> $\mathbf{w}_0 = \mathbf{c}_0 = \mathbf{0}_{L \times 1} ; \gamma_{L,0} = 1 ; \alpha_0 = r_{0,0} = E_0 ; r_{1,0} = 0 ; \rho_{RZAF} = 1 \times 10^{-5}$
--

$$c_0 = 0.01$$

**For n = 1, 2, ... (iterations)**

$$\mathbf{x}_{L,n} = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T$$

**Génération de :  $e_n$**

$$r_{1,n} = \lambda_a r_{1,n-1} + x_n x_{n-1}$$

$$r_{0,n} = \lambda_a r_{0,n-1} + x_n^2$$

$$a_n = \frac{r_{1,n}}{r_{0,n} + c_0}$$

$$e_n = x_n - a_n x_{n-1}$$

$$\alpha_n = \lambda \cdot \alpha_{n-1} + e_n^2$$

**Gain d'adaptation :**

$$\begin{bmatrix} \tilde{\mathbf{c}}_n \\ c_n \end{bmatrix} = \begin{bmatrix} -e_n \\ \lambda \cdot \alpha_{n-1} + c_0 \\ \tilde{\mathbf{c}}_{n-1} \end{bmatrix}$$

$$\delta_n = c_n x_{n-L} + \frac{x_n e_n}{\lambda \cdot \alpha_{n-1} + c_0}$$

$$\gamma_n = \frac{\gamma_{n-1}}{1 + \gamma_{n-1} \delta_n}$$

**Filtrage :**

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

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \bar{\varepsilon}_n \gamma_n \tilde{\mathbf{c}}_n - \rho_{RZA} \frac{\text{sgn}(\mathbf{w}_{n-1})}{1 + \varepsilon_{RZA} |\mathbf{w}_{n-1}|}$$

Table 2. 5: L'algorithme RZA-FNLMS.

## 2.5 Algorithme d'Expectation-Maximisation RLS

Le problème fondamental que nous devons prendre en compte pour obtenir estimation parcimonieuse de  $\mathbf{w}$  est le suivant :

$$\min_{\mathbf{w}} \|\mathbf{w}\|_{\ell_0} \quad \text{Pour } \|d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}\|_{l_2} \leq \delta \quad (2.29)$$

L'équation ci-dessus est un problème classique dans la recherche, et pour résoudre ce problème l'approche commun dans la littérature est de remplacer la norme  $l_0$  par sa meilleure approximation convexe  $l_1$  appelée relaxation convexe, par conséquent la relation (2.29) [20] est remplacée par le problème d'optimisation convexe suivante :

$$\min_{\mathbf{w}} \left\{ \frac{1}{2} \|d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}\|_{l_2}^2 + \rho_{RLS} \|\mathbf{w}\|_{l_1} \right\} \quad (2.30)$$

Où  $\rho_{RLS}$  est le paramètre de régulation.

Au lieu d'utiliser l'équation (2.30) [20] nous nous concentrons sur le problème du maximum de vraisemblance avec pénalité  $l_1$  :

$$\hat{\theta}_{MV} = \arg \max_{\mathbf{w}} \{ \log p(d_n; \mathbf{w}) - \rho_{RLS} \|\mathbf{w}\|_{l_1} \} \quad (2.31)$$

Lorsque les seules données dont on dispose ne permettent pas l'estimation des paramètres et/ou l'expression de la vraisemblance est analytiquement impossible à maximiser, l'algorithme EM peut être une solution, et comme son nom indique cette méthode contient deux étapes : l'étape E (Expectation ; espérance) et l'étape M (Maximisation, maximisation) :

L'étape E : est donnée par l'équation suivante :

$$Q(\theta) = E\{\log p(\theta; \mathbf{w}_n)\} \quad (2.32)$$

L'étape M : procède à la maximisation de la vraisemblance, donnée par l'équation suivante :

$$\hat{\theta}(n) = \arg \max_{\theta} \{ Q(\theta) - \rho_{RLS} \|\mathbf{w}\|_{l_1} \} \quad (2.33)$$

$$\hat{\theta}(n) = \text{sgn}(\Psi_{n-1}) [ |\Psi_{n-1}| - \rho_{EM} \lambda^{-1} \text{diag}(\mathbf{R}_{n-1}^{-1}) ] \quad (2.34)$$



L'algorithme d'EM-RLS est donné ci-dessous dans la table 2.7 [20]:

<p><b>Initialisation :</b>     <math>L = \text{taille de filtre}</math></p> <p><math>\mathbf{w}_0 = \mathbf{k}_0 = \mathbf{0}_{L \times 1}</math> ; <math>\mathbf{R}_0^{-1} = \lambda^{-1} \mathbf{I}_L</math> ; <math>\lambda &lt; 1</math> ; <math>\rho_{EM} = 1 \times 10^{-5}</math></p>
<p><b>Partie filtrage :</b></p> <p><b>For n = 1, 2, ... (iterations)</b></p> <p><math>\mathbf{x}_n = [x_n \quad x_{n-1} \quad \cdots \quad x_{n-L+1}]^T</math></p> $\mathbf{k}_n = \frac{\mathbf{R}_{n-1}^{-1} \mathbf{x}_n}{\lambda + \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1} \mathbf{x}_n}$ <p><math>\bar{\varepsilon}_n = d_n - \mathbf{x}_n^T \mathbf{w}_{n-1}</math></p> <p><math>\Psi_n = \Psi_{n-1} + \bar{\varepsilon}_n \mathbf{k}_n</math></p> $\mathbf{R}_n^{-1} = \lambda^{-1} [\mathbf{R}_{n-1}^{-1} - \mathbf{k}_n \mathbf{x}_n^T \mathbf{R}_{n-1}^{-1}]$ <p><math>\mathbf{w}_n = \text{sgn}(\Psi_{n-1}) [ \Psi_{n-1}  - \rho_{EM} \lambda^{-1} \text{diag}(\mathbf{R}_{n-1}^{-1})]_+</math></p> <p>Où <math>[\ ]_+</math> désigne le <math>\max[z, 0]</math>.</p>

Table 2. 6: L'algorithme EM-RLS.

## 2.6 Conclusion

Dans ce chapitre, nous avons présenté les algorithmes parcimonieux de type gradient stochastique (ZA-NLMS, RZA-NLMS), ainsi que la version de type NLMS rapide (FNLMS). Ensuite, nous avons présenté deux nouveaux algorithmes rapides type Fast NLMS améliorés pour l'identification de réponses impulsionnelles parcimonieuses (R)ZA-FNLMS avec complexité comparable à l'algorithme NLMS et une vitesse de convergence rapide. A la fin nous avons décrit aussi l'algorithme d'expectation-maximisation de RLS. Les résultats des simulations et les comparaisons seront présentés dans le troisième chapitre.

## Chapitre 3 Résultat de simulations

---

### 3.1 Mesure du degré de parcimonie

Il existe plusieurs formes des réponses impulsionnelles selon les types de salles (salle conférence, audio, studio ... etc.), et leurs caractéristiques dépendent de plusieurs facteurs. Dans cette partie nous discutons sur la mesure du degré de parcimonie d'un vecteur.

#### 3.1.1 La norme d'un vecteur

Il y a plusieurs normes dans littérature, mais considérons, seulement quatre normes pour notre mesure.

On considère le vecteur :

$$w = [w_0 \quad w_1 \quad \cdots \quad w_{L-1}]^T \neq 0 \quad (3.1)$$

De taille  $L > 1$ , on définit la fonction :

$$f(w_l) = \begin{cases} 1, & w_l \neq 0 \\ 0, & w_l = 0 \end{cases} \quad (3.2)$$

Puis la norme  $l_0$  de  $w$  est donnée par :

$$\|w\|_0 = \sum_{l=0}^{L-1} f(w_l) \quad (3.3)$$

Autrement dit, la norme  $l_0$  d'un vecteur est égale au nombre de ses éléments non nuls, pour  $w \neq 0$ , on a toujours :

$$1 \leq \|w\|_0 \leq L$$

### 3.1.2 Mesure du degré de la parcimonie

La fonction  $f(w_l)$  n'est pas continue, la plupart des coefficients de  $w$  ont des valeurs très faibles, mais pas forcément nulles, ce qui fait que la norme  $l_0$  dans (3.3) devient difficile à utiliser pratiquement et c'est souvent une approximation par une fonction continue [21].

Les normes  $l_1$ ,  $l_2$  et  $l_\infty$  (maximum) de vecteur  $w$  sont définies par :

$$\|w\|_1 = \sum_{l=0}^{L-1} |w_l| \quad (3.4)$$

$$\|w\|_2 = \sqrt{\sum_{l=0}^{L-1} w_l^2} = \sqrt{w^T w} \quad (3.5)$$

$$\|w\|_\infty = \max_{0 \leq l \leq L-1} |w_l| \quad (3.6)$$

La parcimonie d'une réponse impulsionnelle notée par  $\xi(w)$  qui est une valeur comprise entre 0 et 1 :

$$0 \leq \xi(w) \leq 1 \quad (3.7)$$

Alors le degré de parcimonie est mesuré par :

$$\xi(w) = \frac{L}{L - \sqrt{L}} \left( 1 - \frac{\|w\|_1}{\sqrt{L}\|w\|_2} \right) \quad (3.8)$$

Quelques propriétés de cette mesure :

1.  $\forall a \neq 0, \xi(aw) = \xi(w)$
2.  $\xi(w_d) = 1.$
3.  $\xi(w_u) = 0.$

Où  $w_d$  est le filtre de Dirac, et  $w_u$  est le filtre d'échelon (uniforme).

### 3.2 Génération synthétique de RI parcimonieuse

Dans ce chapitre nous allons présenter quelques résultats de simulation de RI tester pour les performances des algorithmes étudiés, ces algorithmes sont testés avec des RI parcimonieuses dont la méthode de génération est donnée dans [22] comme suit :

Soit le vecteur  $u$  :

$$u_{L \times 1} = [\mathbf{0}_{L_p \times 1} \quad 1 \quad e^{-1/\psi} \quad e^{-2/\psi} \quad \dots \quad e^{-(L_u-1)/\psi}]^T \quad (3.9)$$

Où  $L_p$  est la longueur du vecteur des zéros, et  $L_u = L - L_p$  est la longueur de la fenêtre décroissante tandis que  $\psi \in \mathbb{Z}^+$  est la constante de décroissance qui peut varier dans le temps.

Pour la brièveté de la notation, nous ignorons la dépendance de  $\psi$  sur l'index temporel  $n$ .

En définissant un vecteur  $L_u \times 1$  de  $b$  comme une séquence de bruit blanc gaussien de moyenne nulle (WGN) avec la variance  $\sigma_b^2$ , la réponse impulsionnelle synthétique  $L \times 1$  peut alors être exprimées comme suit :

$$B_{L_u \times L_u} = \text{diag}(b) \quad (3.10)$$

$$w_n = \begin{bmatrix} 0_{L_p \times L_p} & 0_{L_p \times L_u} \\ 0_{L_u \times L_p} & B_{L_u \times L_u} \end{bmatrix} u + p \quad (3.11)$$

Où  $p$  est un vecteur qui s'assure que les  $L$  éléments de la région « inactive » sont petits, mais non nuls et qu'il s'agit d'une séquence WGN de moyenne nulle indépendante avec la variance  $\sigma_p^2$ .

### 3.3 Signaux de la simulation

Dans cette étude nous avons simulé les algorithmes décrits dans le chapitre précédent avec deux types de signaux d'entrées stationnaires et non-stationnaires, filtrés par différents degrés de parcimonie de RI synthétiques. D'abord, nous présentons les RI avec différents degrés de parcimonie utilisant l'approche de section 3.1 (voir les figures 3.1), puis nous montrons trois RI acoustiques, la première dans un habitacle de la voiture, puis la RI dans une salle audioconférence ainsi la RI dans une salle visioconférence (voir les figures 3.2).

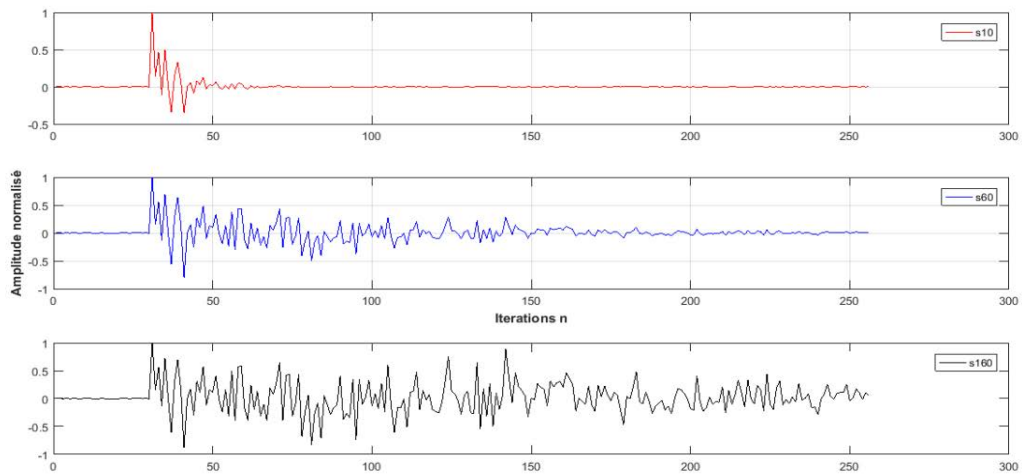


Figure3. 1:RI synthétique avec  $L=256$ ,  $L_p=30$ ,  $\psi=10,60,160$ .

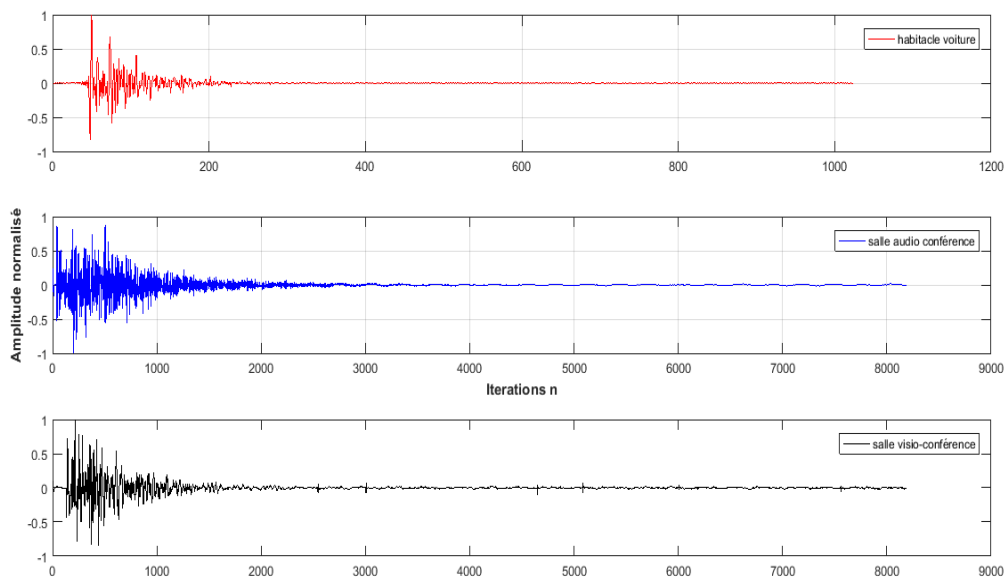


Figure3. 2: RI réel avec  $L=256$ , d'un habitacle de voiture, d'une salle audioconférence et d'une salle visio-conférence.

Concernant le signal d'entrée, nous avons utilisé au début un signal stationnaire USASI dont son spectre ressemble en moyenne au spectre de signal parole (voir figure 3.3), puis nous avons utilisé la parole comme un signal non-stationnaire constitué deux phrases (d'un homme et d'une femme) échantillonnées sur 16 kHz et codé sur 16 bit, présentés dans la figure 3.4.

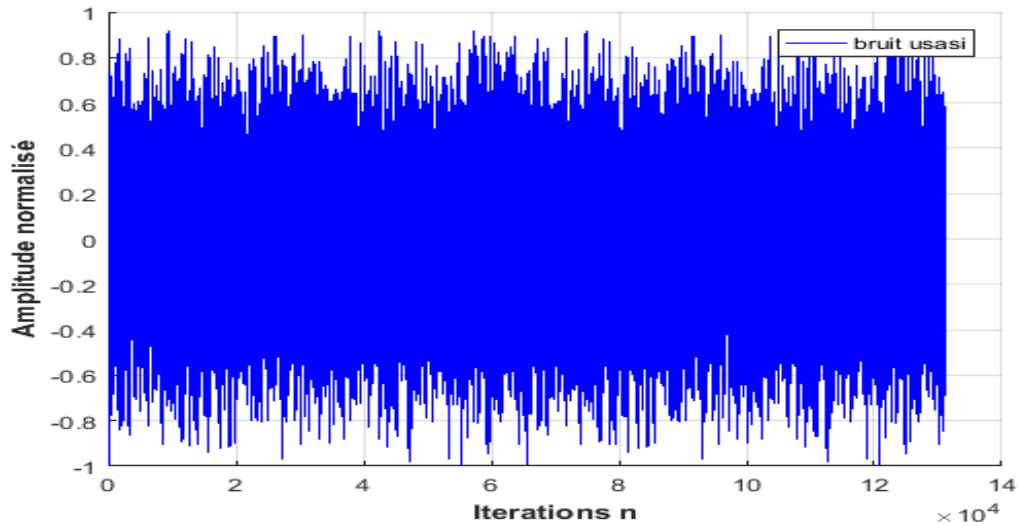


Figure3. 3: signal d'entrée stationnaire USASI.

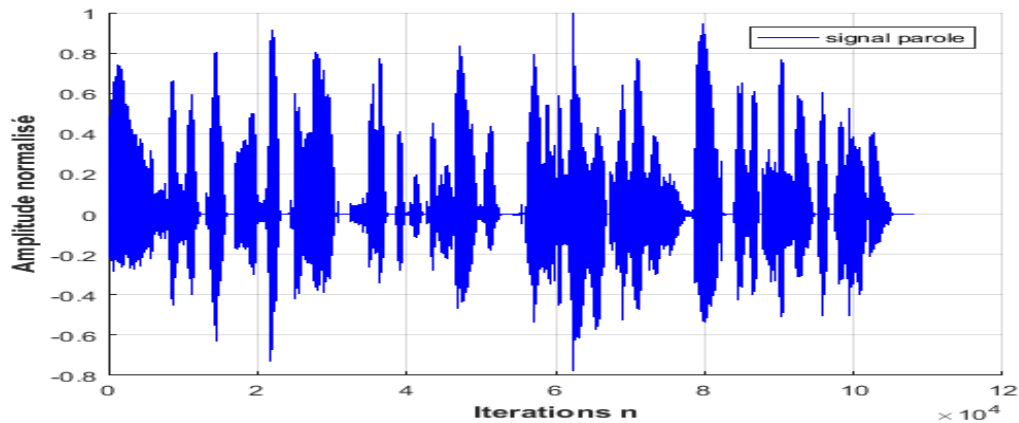


Figure3. 4: signal d'entrée non-stationnaire (la parole).

### 3.4 Description des critères de performance :

- Pour comparer les performances des algorithmes entre eux, un critère couramment utilisé est L'EQM (MSE) donnée par :

$$MSE(dB) = 10 \log_{10}(\langle e^2(n) \rangle)$$

$\langle . \rangle$  Moyenne de 256 échantillons.

- Nous avons ajouté le bruit blanc au signal désiré avec un rapport signal sur bruit (RSB) donnée pour voir le comportement des algorithmes dans ce cas.
- Pour les valeurs de pas d'adaptation  $\mu$ , nous l'avons le fixé à un pour donner une performance meilleure.

- Dans toutes nos simulations, nous utilisons les 50 premiers blocs de 256 points, pour assurer l'apparition de la convergence initiale et l'erreur en régime permanent.

### 3.5 Simulation des algorithmes

Dans ce qui suit nous allons présenter les résultats des simulations des algorithmes de filtrage adaptatif pour l'identification de RI dans le cadre de l'annulation d'écho acoustique, pour cela nous allons partager nos simulations en deux catégories : première catégorie concerne les algorithmes de RZA-NLMS, ZA-NLMS et NLMS et ces versions rapides, la deuxième catégorie concerne les algorithmes de moindres carrés récursif EM-RLS et RLS décrit dans la section 2.5.

### 3.6 Simulation des algorithmes (R)ZA-NLMS et NLMS

Dans cette partie, nous allons présenter les résultats de simulation d'algorithmes mentionnés dans le titre, utilisant des RI synthétiques et réel avec différents niveaux de parcimonie où le signal d'entrée est stationnaire (USASI) et non-stationnaire (la parole), mais tout d'abord nous montrons le choix des paramètres  $\rho_{(R)ZA}$  et  $\varepsilon_{RZA}$  et quelles sont les valeurs qui donnent les meilleures performances des algorithmes (R)ZA-NLMS, pour un signal d'entrée USASI et RI parcimonieuse.

#### ***a*** Le choix de paramètre $\rho_{(R)ZA}$ :

Pour assurer une bonne performance des algorithmes (R)ZA-NLMS, nous devons mieux choisir le pas de régulation  $\rho_{(R)ZA}$ , pour cela nous allons utiliser une RI parcimonieuse avec  $\psi = 10$  et  $L=256$ , le résultat de simulation donné dans la figure 3.5.

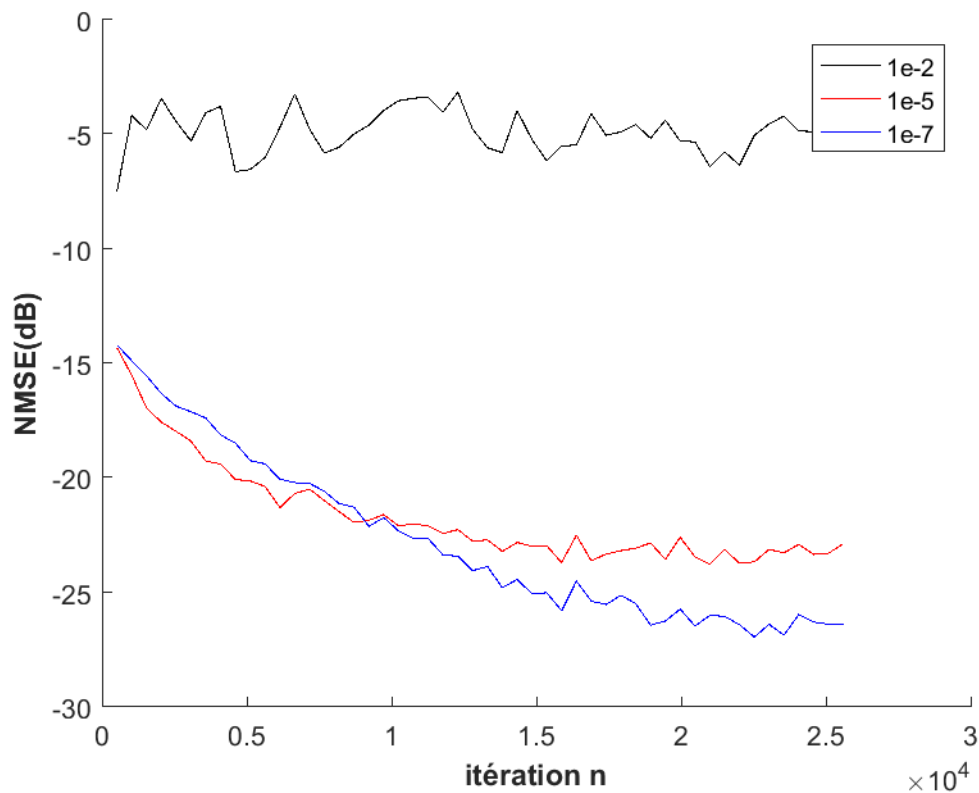


Figure3. 5: l'effet de  $\rho_{(R)ZA}$  sur le comportement de l'algorithme ZA-NLMS, SNR=30.

D'après la figure 3.5 on remarque que pour une faible valeur de  $\rho_{(R)ZA}$  d'ordre  $10^{-7}$  l'algorithme de ZA-NLMS donne les mêmes performances que le NLMS, par contre une très grande valeur de  $\rho_{(R)ZA}$  (supérieure à  $10^{-2}$ ) nous donne une divergence de l'algorithme de ZA-NLMS, et pour une valeur égale à  $10^{-5}$  l'algorithme de ZA-NLMS converge rapidement que les autres valeurs de  $\rho_{(R)ZA}$ .

#### **b Le choix de paramètre $\varepsilon_{RZA}$**

La figure 3.6 nous permet de déduire la valeur de  $\varepsilon_{RZA}$ , alors pour une faible valeur de  $\varepsilon_{RZA}$  nous avons une faible convergence de l'algorithme RZA-NLMS par rapport aux autres valeurs, d'autre part si  $\varepsilon_{RZA}$  est grand l'algorithme converge lentement, mais donne une bonne erreur finale, et  $\varepsilon_{RZA}=50$  est la valeur optimum qui donne le bon fonctionnement de l'algorithme RZA-NLMS.



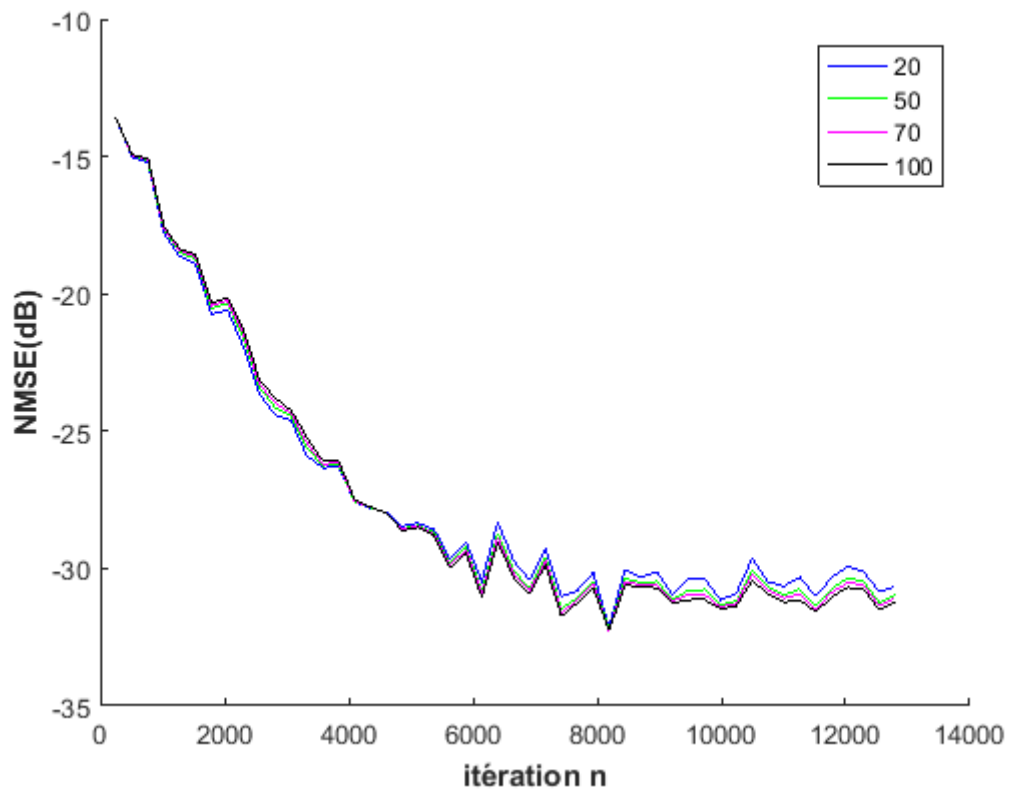


Figure3. 6: l'effet de  $\epsilon_{RZA}$  sur le fonctionnement de l'algorithme RZA-NLMS.

### 3.6.2 Cas d'entrée stationnaire (USASI)

Maintenant nous montrons les résultats des simulations que nous avons réalisées utilisant le signal USASI filtré par les RI synthétiques dispersive ( $\psi = 60$ ) et parcimonieuse ( $\psi = 10$ ), avec un SNR=30dB (voir figures 3.7, 3.8).

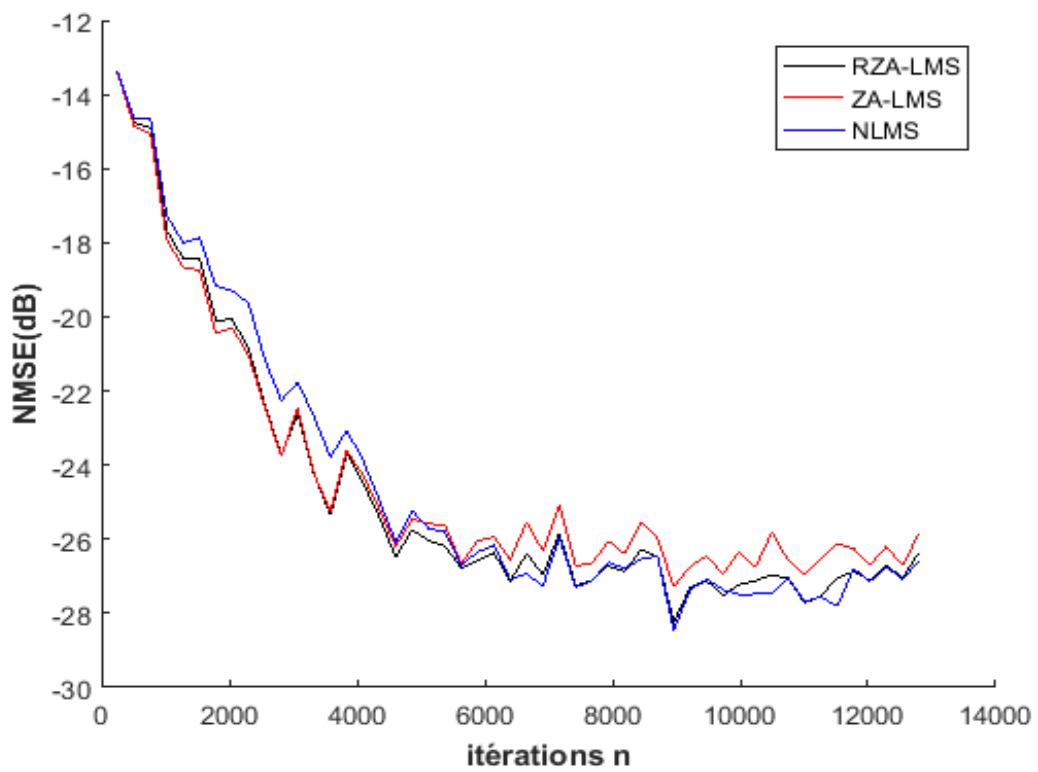


Figure3. 7: signal d'entré USASI, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

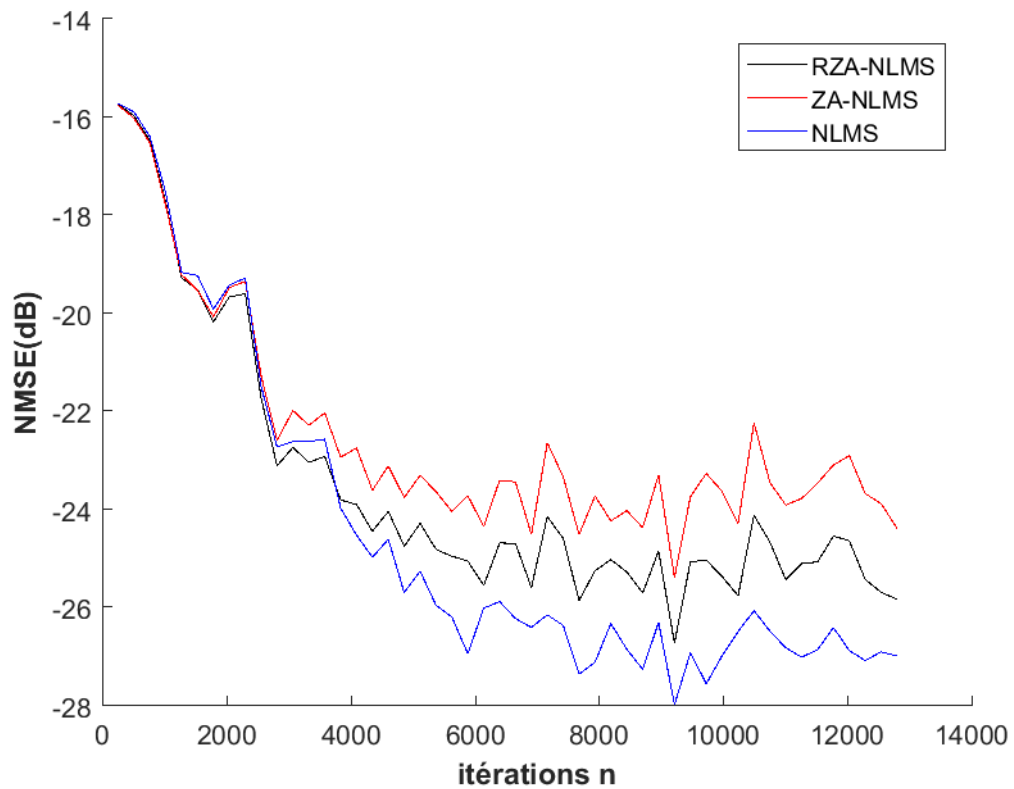


Figure3. 8: signal d'entré USASI, RI synthétique avec  $L=256$ ,  $\psi=60$ ,  $\mu=1$ ,  $SNR=30dB$ .

D'après les figures 3.7 et 3.8 on remarque que les algorithmes de RZA-NLMS et ZA-NLMS ont une convergence rapide et une erreur régime permanent réduite avec la RI parcimonieuse que la RI dispersive.

Maintenant nous allons varier le SNR pour voir le comportement de ces algorithmes et leurs performances, nous gardons la même taille  $L=256$  et le même degré de parcimonie  $\psi = 10$ , avec signal d'entrée USASI et RI synthétique (voir les figures 3.9,3.10,3.11).

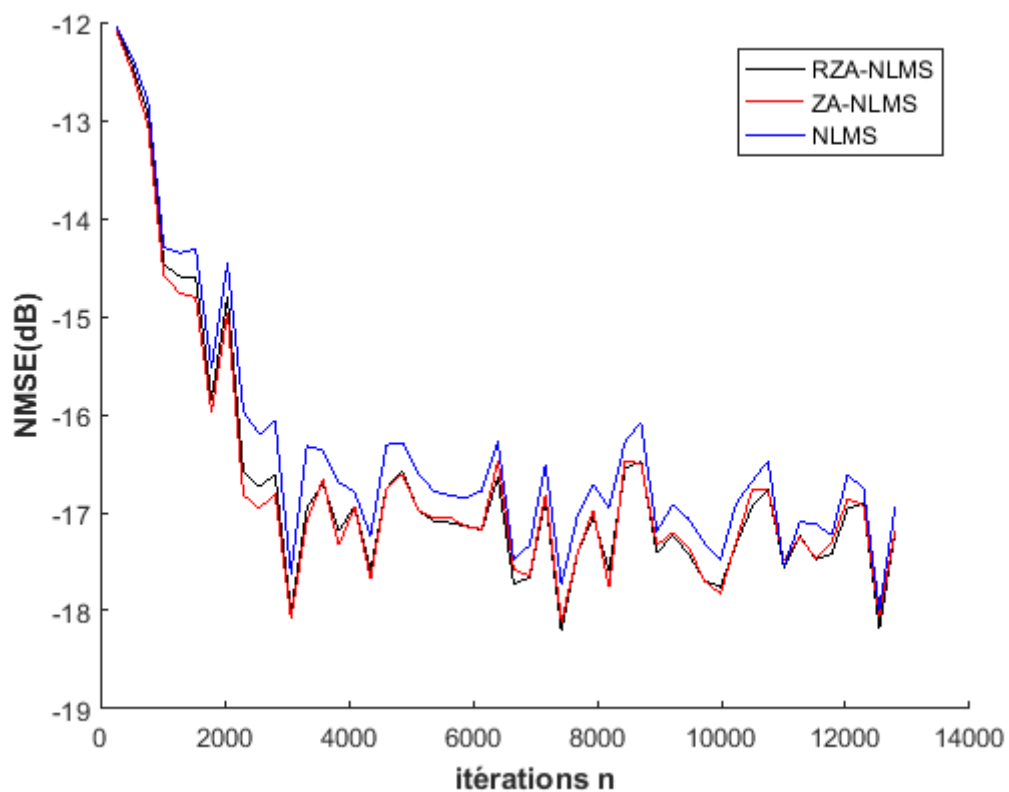


Figure3. 9: signal d'entrée USASI, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=20dB$ .

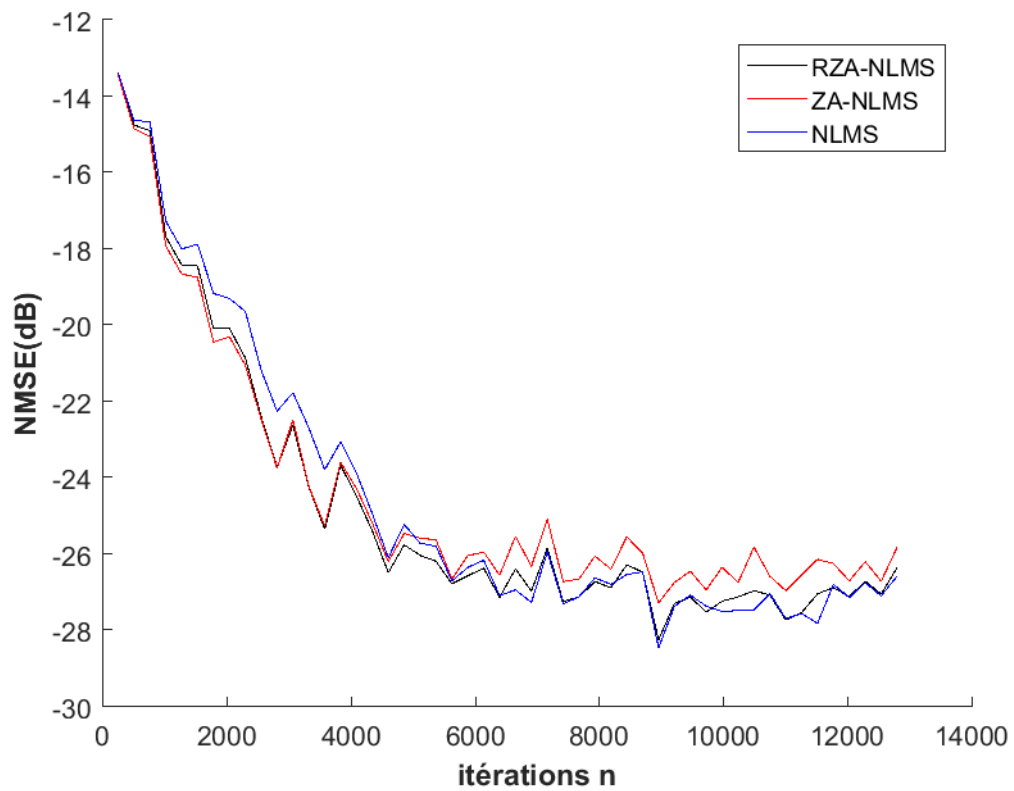


Figure3. 10: signal d'entrée USASI, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

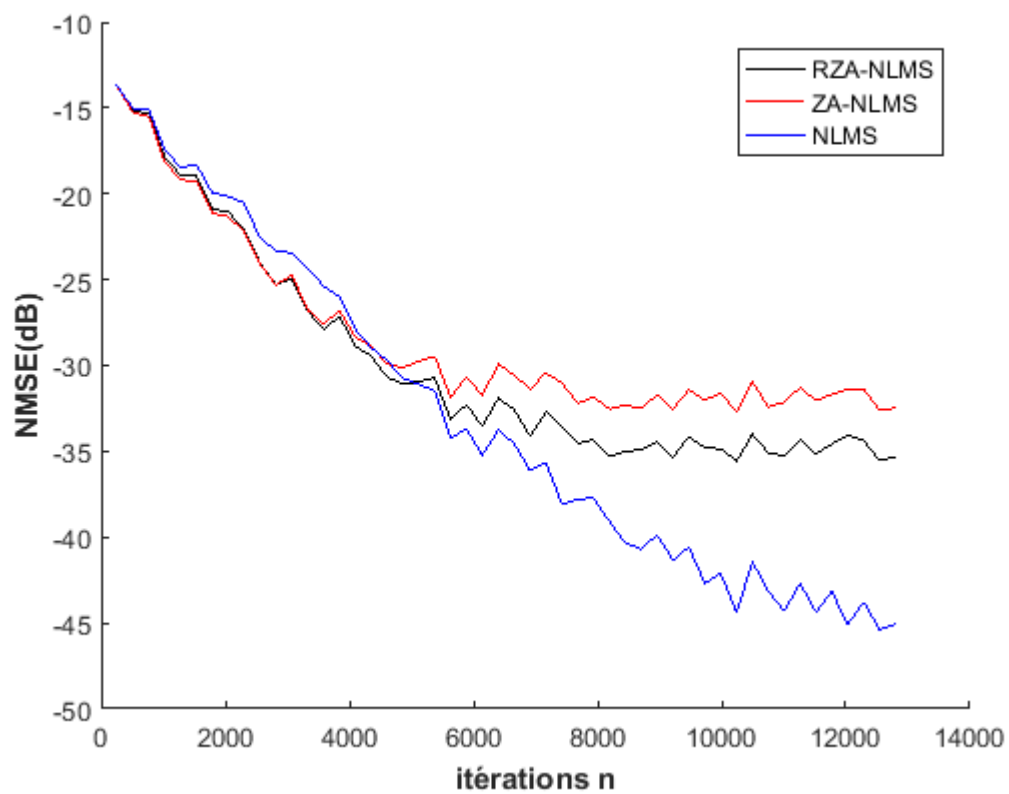


Figure3. 11: signal d'entrée USASI, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=50dB$ .

Les figures 3.9, 3.10 et 3.11 montrent que les algorithmes de RZA-NLMS et ZA-NLMS ont une vitesse de convergence rapide quelle que soit la valeur de SNR, par contre l'erreur en régime permanent de ces deux algorithmes est meilleure par rapport à l'algorithme NLMS avec un SNR égal à 20dB ou 30dB.

Nous observons dans les figures 3.12, 3.13 et 3.14 que lorsque nous augmentons la taille de filtre L la vitesse de convergence de RZA-NLMS est meilleure que ZA-NLMS et NLMS, mais en régime permanent l'erreur de NLMS est plus faible que les algorithmes RZA-NLMS et ZA-NLMS.

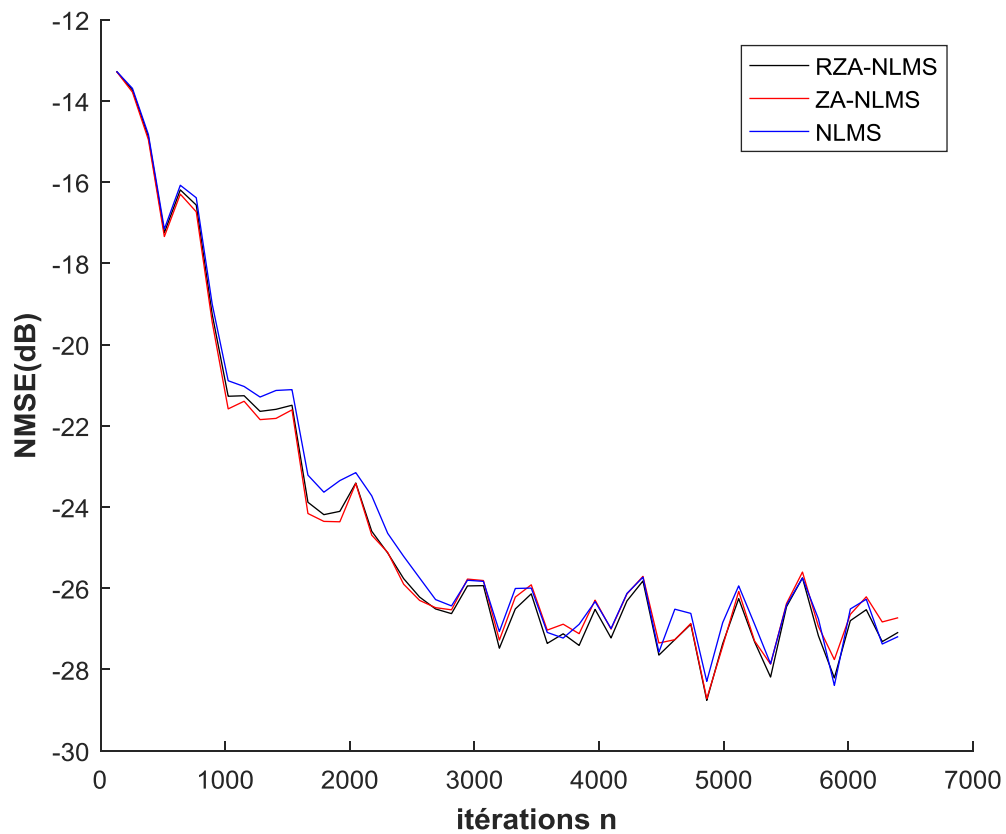


Figure 3. 12: signal d'entrée USASI, RI synthétique (256) avec  $L=128$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

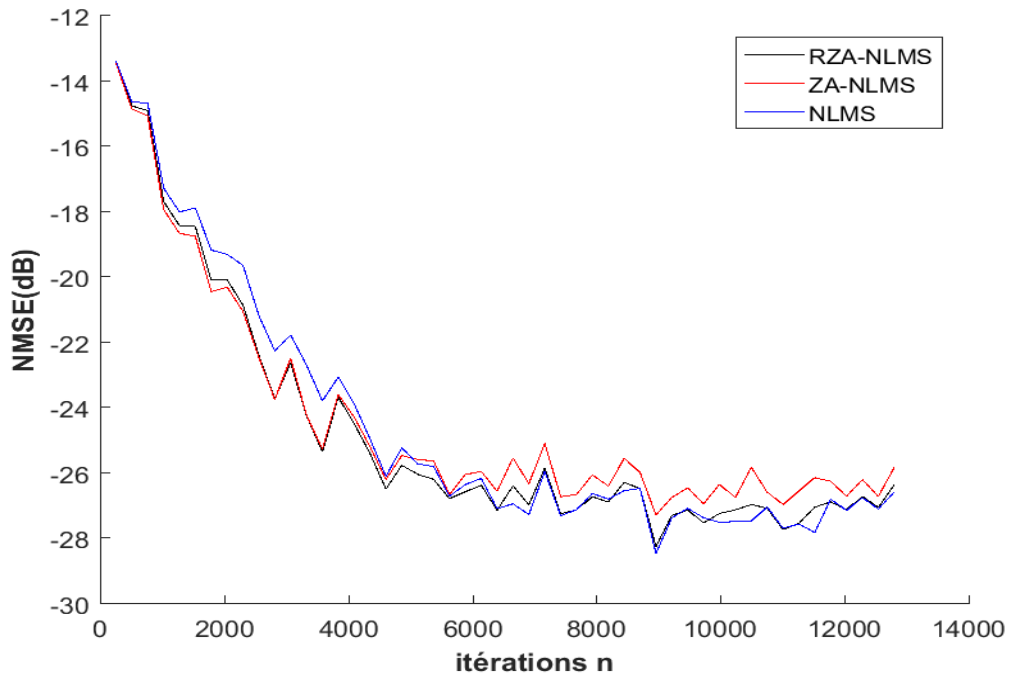


Figure3. 13 signal d'entrée USASI, RI synthétique (256) avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

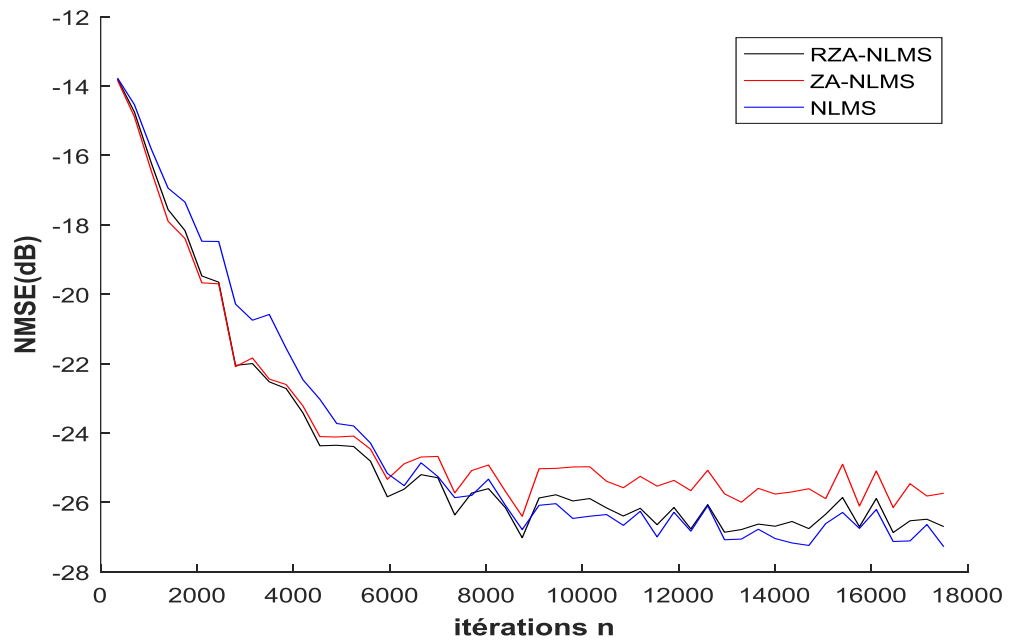


Figure3. 14: signal d'entrée USASI, RI synthétique (256) avec  $L=350$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

Nous avons utilisé la RI réel du l'habitable d'une voiture avec l'entrée USASI et un SNR=30dB, pour différentes tailles (L=256 et 350), les résultats sont donnés par les figures 3.15 et 3.16.

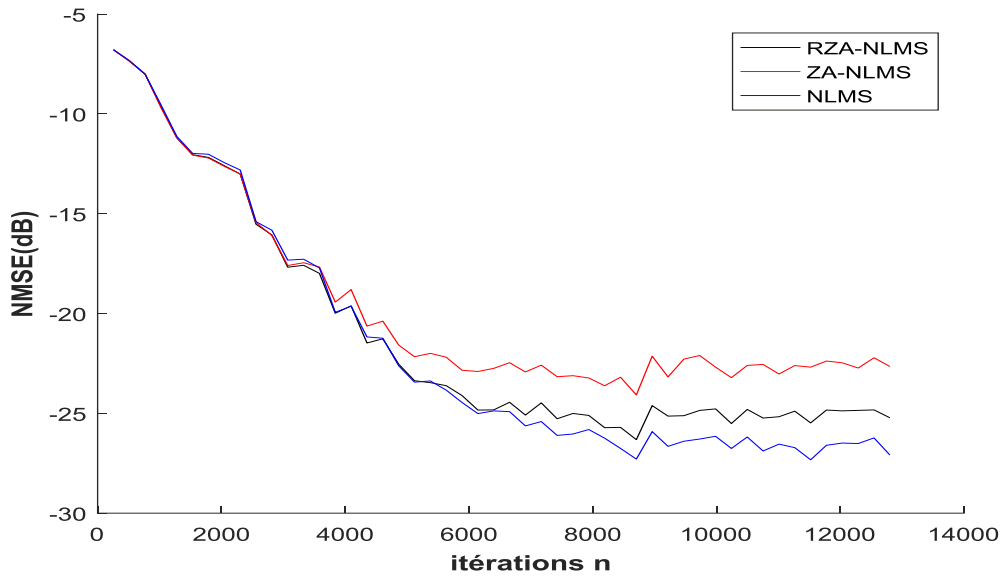


Figure3. 15: signal d'entrée USASI, RI d'un habitacle d'une voiture (256) avec L=256,  $\mu=1$ , SNR=30dB.

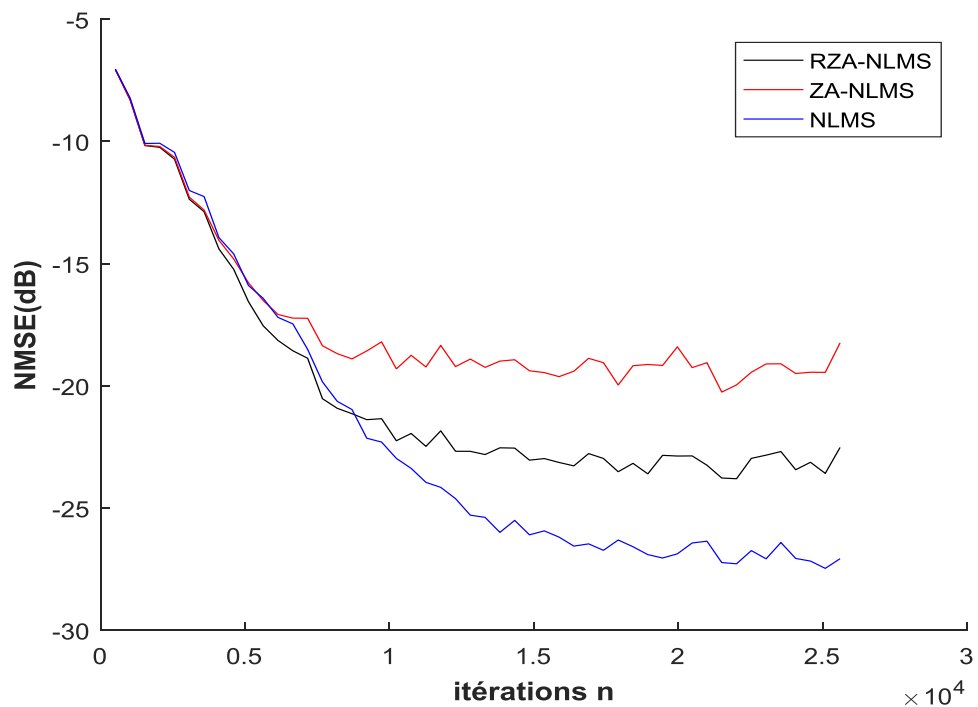


Figure3. 16: signal d'entrée USASI, RI d'un habitacle d'une voiture (256) avec L=350,  $\mu=1$ , SNR=30dB.

D'après ces figures ci-dessus on remarque que l'algorithme de RZA-NLMS converge rapidement si la taille égale à 350.

### 3.6.3 Cas d'entrée non-stationnaire (parole)

Les simulations suivantes, utilisent la parole et des RI parcimonieuses et dispersives synthétiques avec des tailles de filtres différents et des SNR de 30dB et 50 dB.

Dans les figures 3.17 et 3.18 les algorithmes de RZA-NLMS et ZA-NLMS sont plus performants avec la RI parcimonieuse que la RI dispersive devant l'algorithme NLMS.

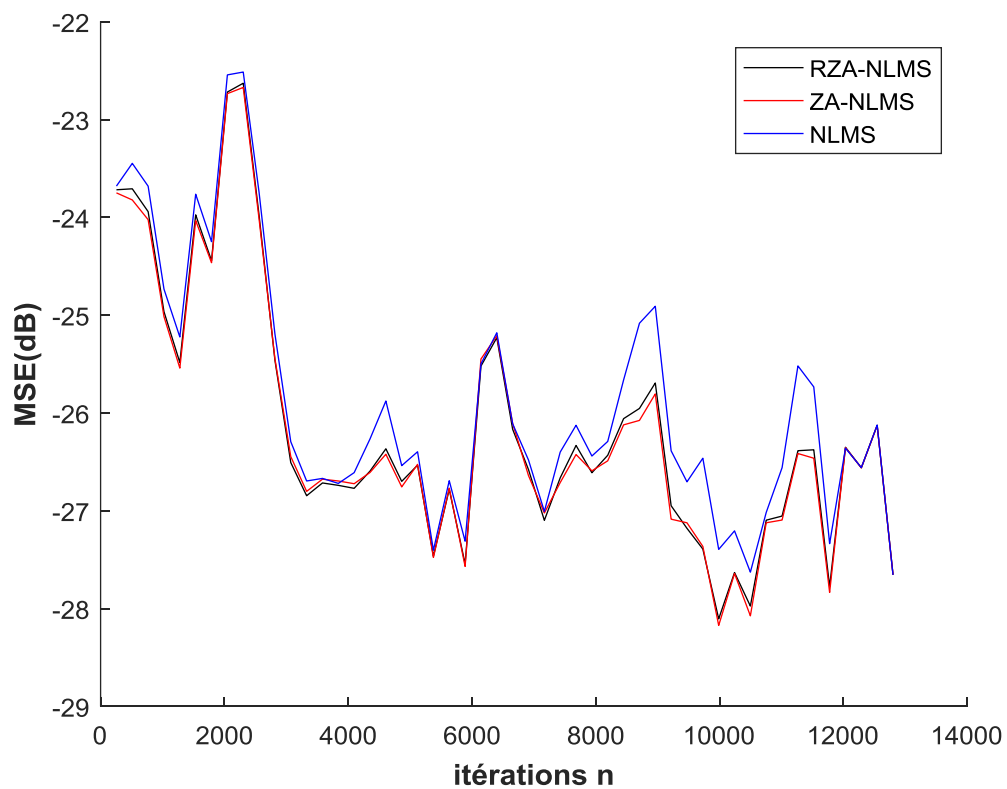


Figure3. 17: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ , SNR=30dB.



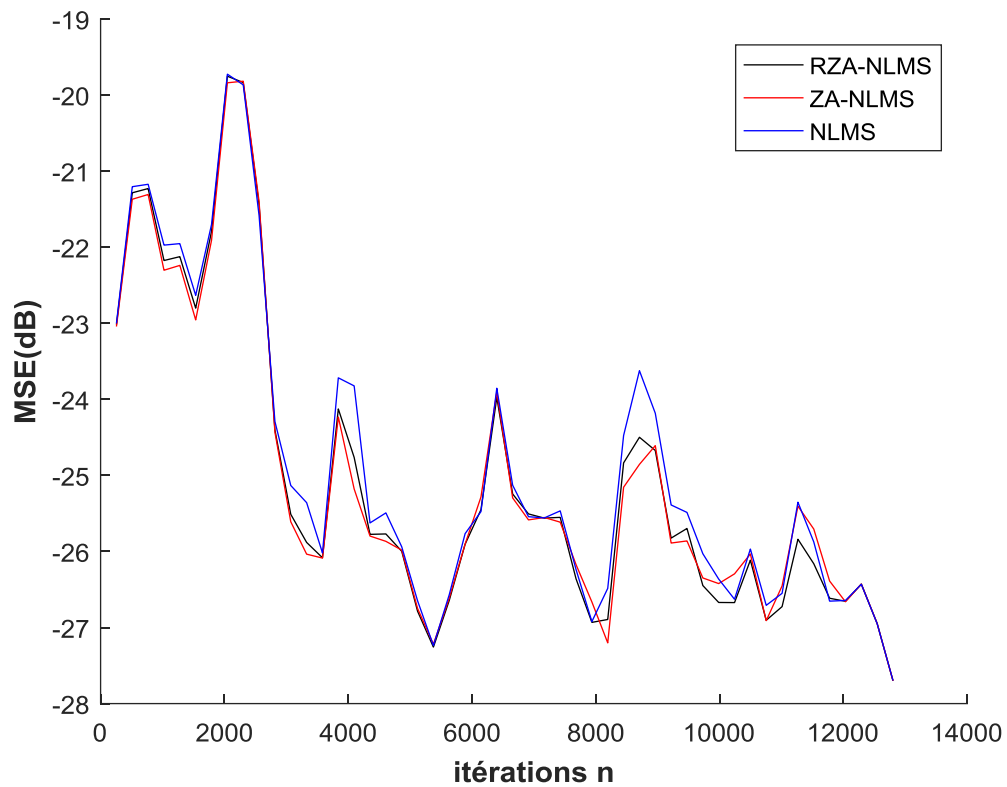


Figure3. 18: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=256$ ,  $\psi=60$ ,  $\mu=1$ ,  $SNR=30dB$ .

A partir des résultats de simulations présentés dans les figures 3.19,3.20,3.21 et 3.22, on remarque que les algorithmes de RZA-NLMS et ZA-NLMS ont une bonne performance par rapport au l'algorithme NLMS quel que soit le SNR avec une RIP.

Avec les RI dispersives l'algorithme NLMS devient plus performant avec des valeurs de SNR élevés ( $SNR=50dB$ ).

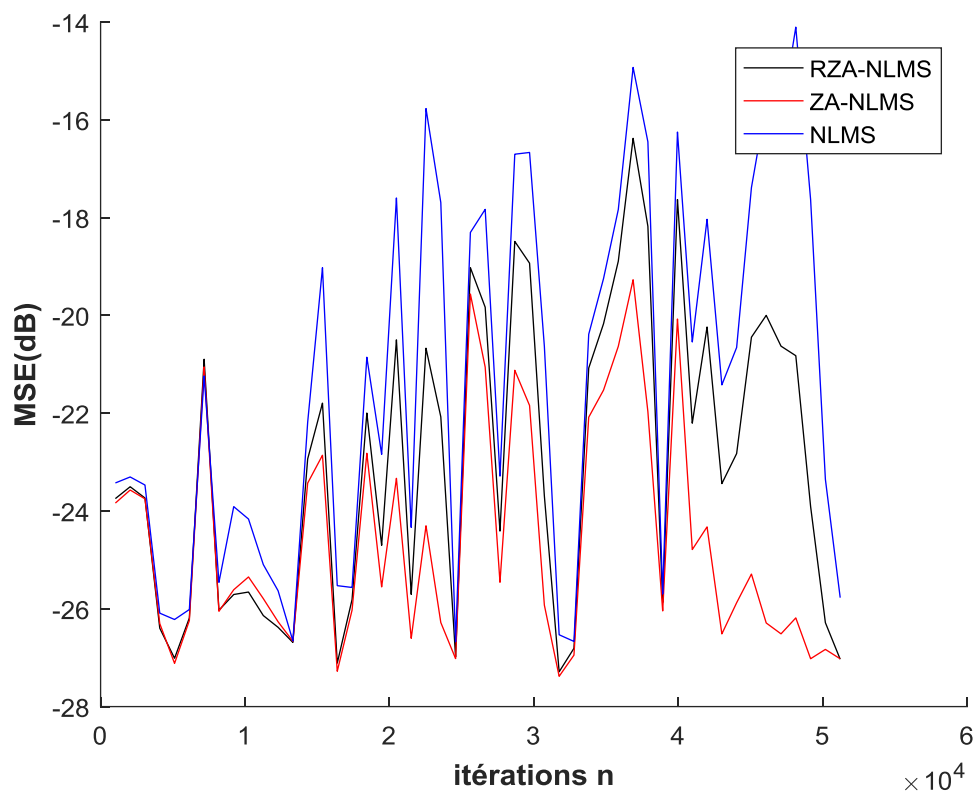


Figure3. 19: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=1024$ ,  $\psi=10$ ,  $\mu=1$ , SNR=30dB.

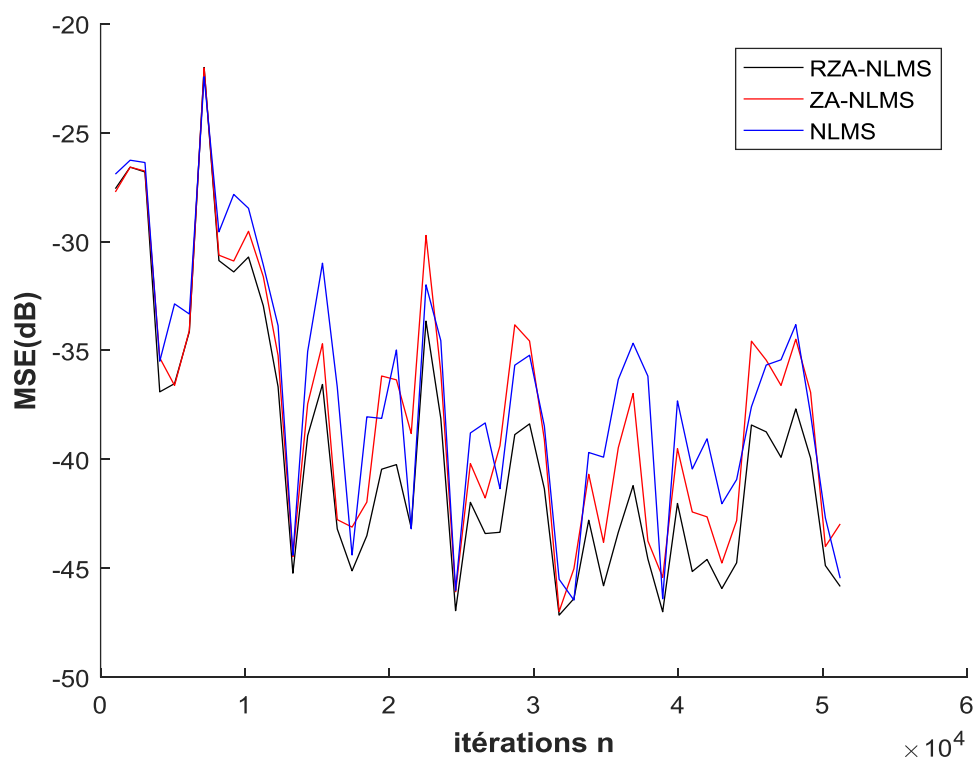


Figure3. 20 signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=1024$ ,  $\psi=10$ ,  $\mu=1$ , SNR=50dB.

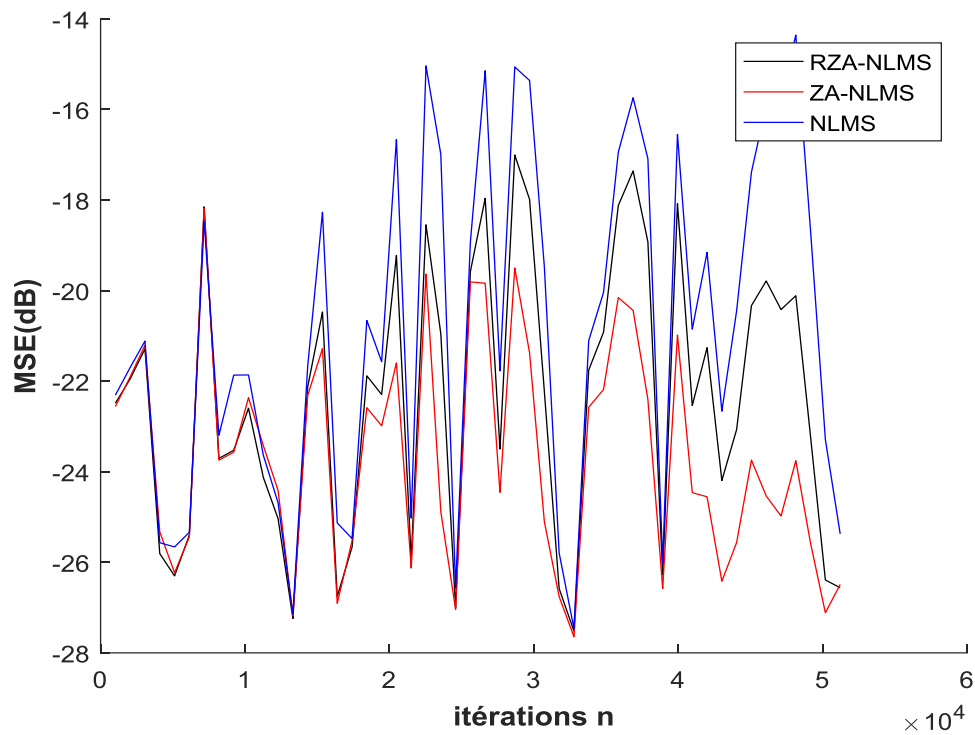


Figure3. 21: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=1024$ ,  $\psi=60$ ,  $\mu=1$ , SNR=30dB.

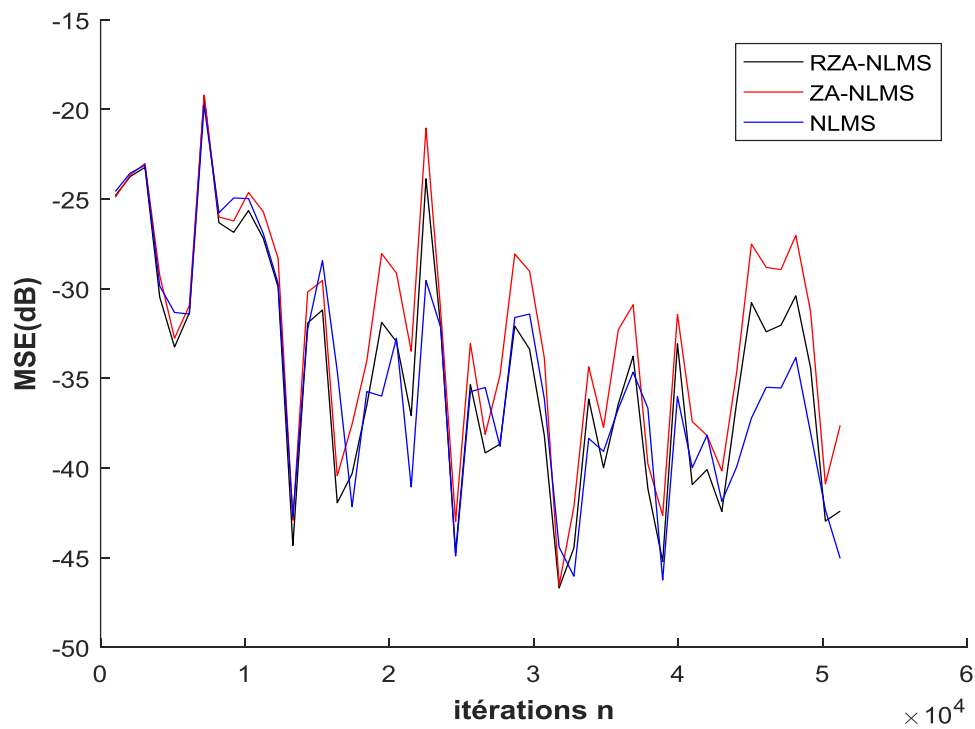


Figure3. 22: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=1024$ ,  $\psi=60$ ,  $\mu=1$ , SNR=50dB.

### 3.7 Simulation des algorithmes (R)ZA-FNLMS et FNLMS

Pour cette catégorie d'algorithmes, nous allons présenter les résultats des simulations avec RIP et RID avec différentes valeurs du SNR, nous commençons par des signaux stationnaires (USASI) puis avec un signal non-stationnaire (parole).

#### 3.7.1 Cas d'un signal stationnaire (USASI)

D'après les résultats obtenus dans les figures ci-dessous, on peut dire que l'algorithme RZA-FNLMS converge rapidement avec des RIP et RID pour la taille de filtre égale à 256 ou 1024 points quelle que soit la valeur de SNR, mais l'erreur en régime permanent l'algorithme FNLMS est meilleure que les algorithmes RZA-FNLMS et ZA-FNLMS pour un longueur de filtre  $L=256$ , par contre lorsqu'on augmente la taille de filtre les algorithmes RZA-FNLMS et ZA-FNLMS devient plus performant en terme d'erreur en régime permanent que l'algorithme FNLMS.

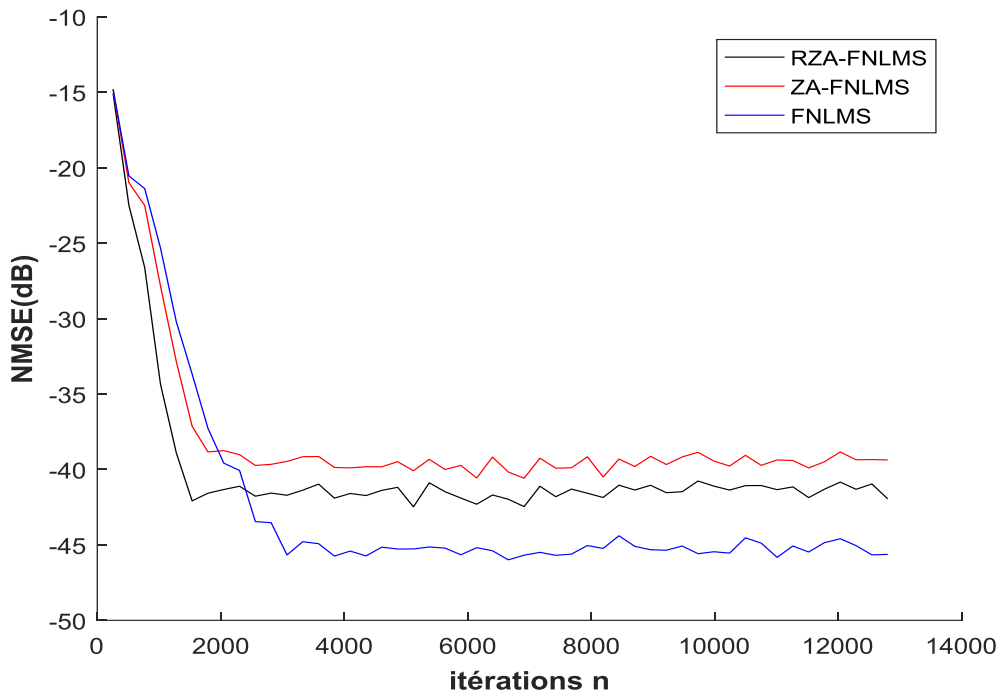


Figure3. 23: signal d'entrée USASI, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=50dB$ .

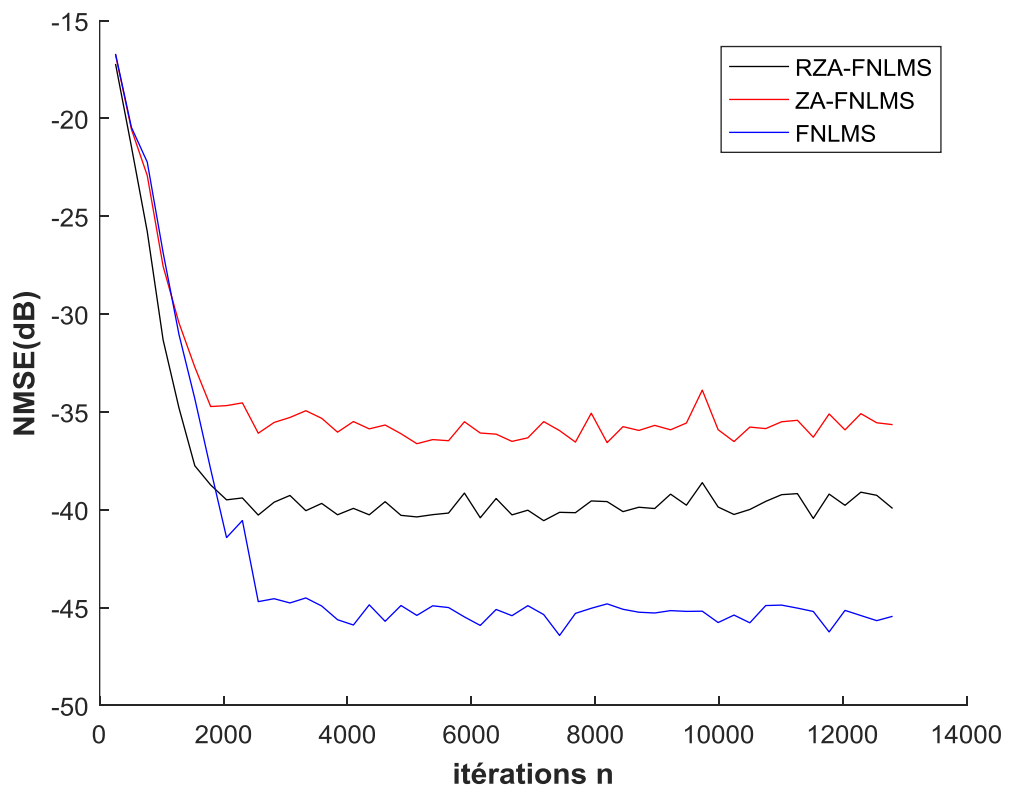


Figure3. 24: signal d'entrée USASI, RI synthétique avec  $L=256$ ,  $\psi=60$ ,  $\mu=1$ ,  $SNR=50dB$ .

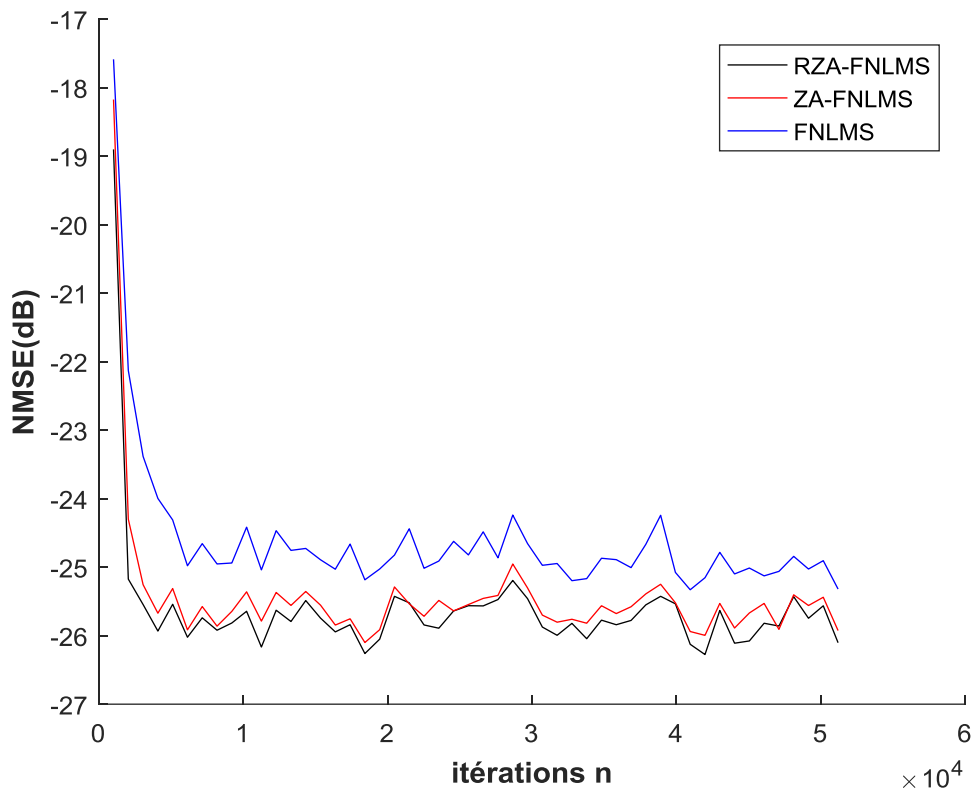


Figure3. 25: signal d'entrée USASI, RI synthétique avec  $L=1024$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

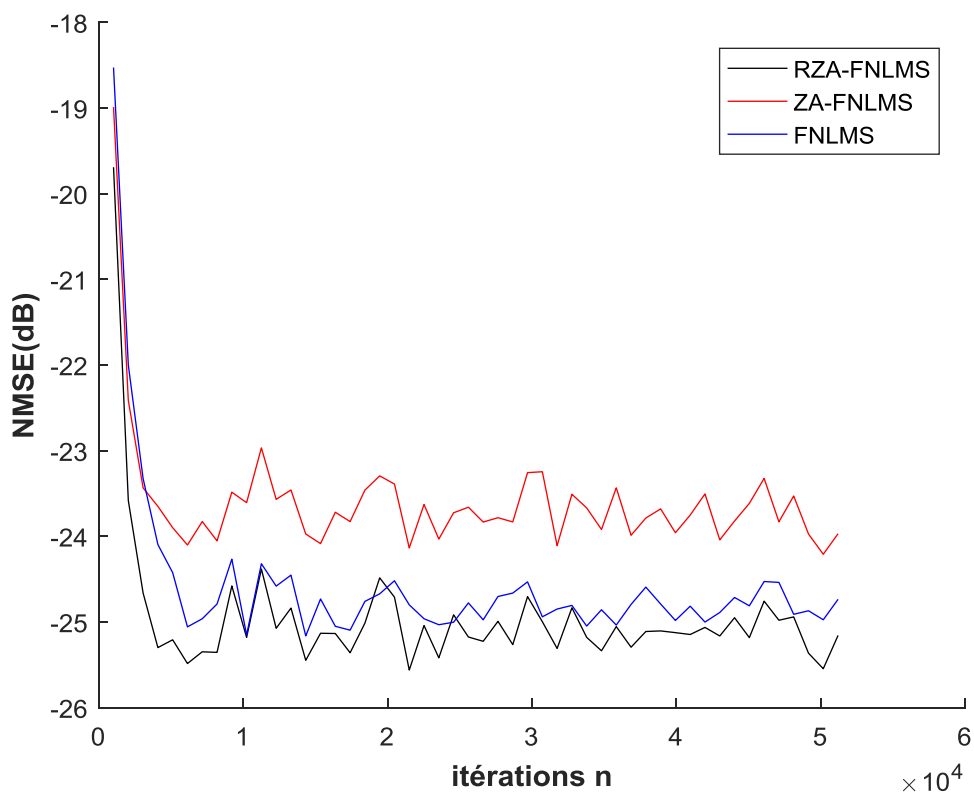


Figure3. 26: signal d'entrée USASI, RI synthétique avec  $L=1024$ ,  $\psi=60$ ,  $\mu=1$ ,  $SNR=30dB$ .

### 3.7.2 Cas d'un signal non-stationnaire (parole)

Pour voir le comportement des algorithmes FNLMS avec les signaux non-stationnaires, nous avons utilisé la parole comme signal d'entrée avec des RIP et RID et des valeurs de SNR différents, les expériences ci-dessous sont réalisées avec des tailles de filtre 256, 512 et 1024 points.

D'après les figures (3.27, 3.28 et 3.29) nous observons que l'algorithme RZA-FNLMS devient plus performant si nous augmentons le SNR, mais ses performances se dégradent avec la RI dispersive.

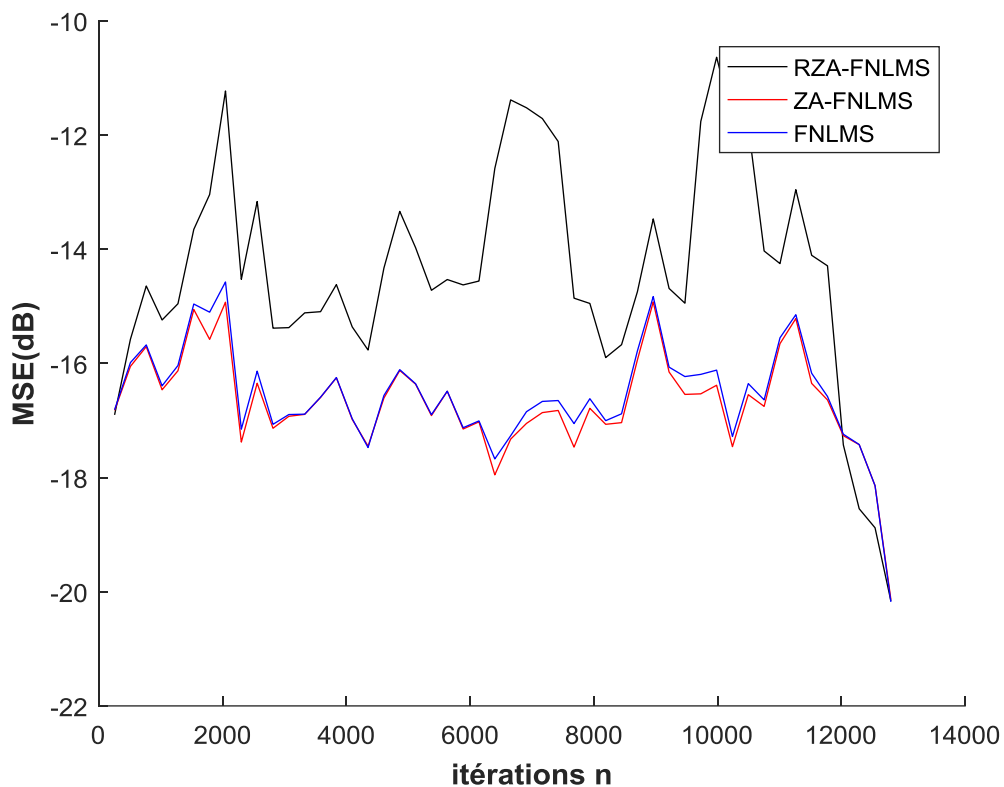


Figure3. 27: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=20dB$ .

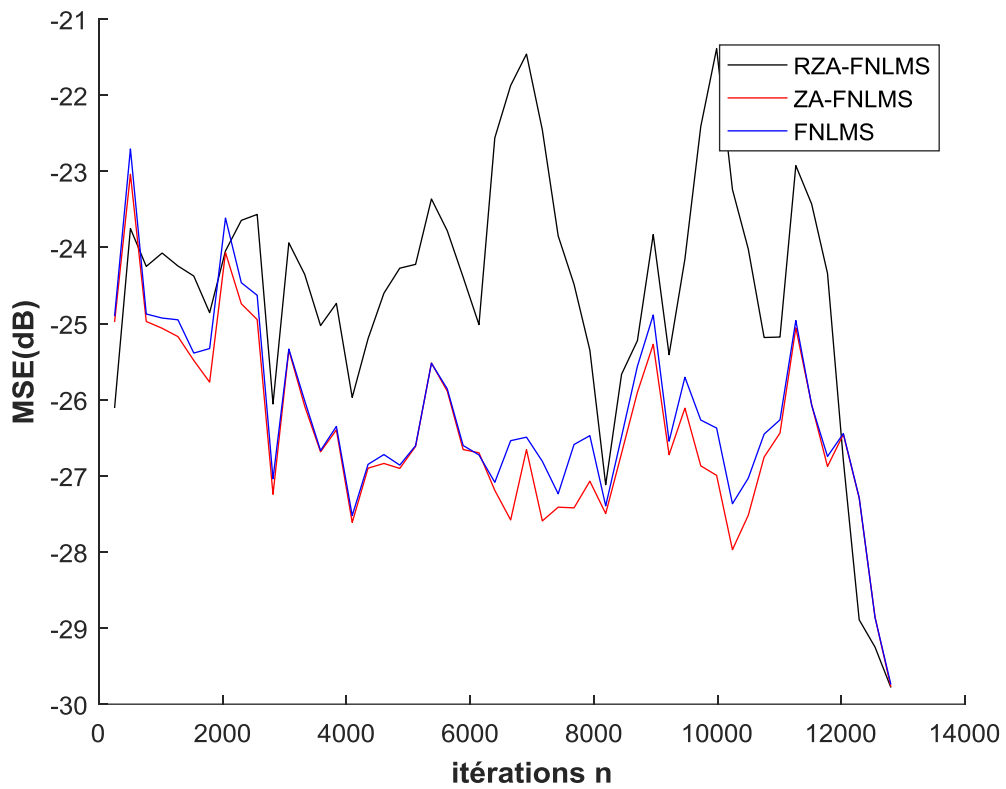


Figure3. 28: signal d'entrée Parole, RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=30dB$ .

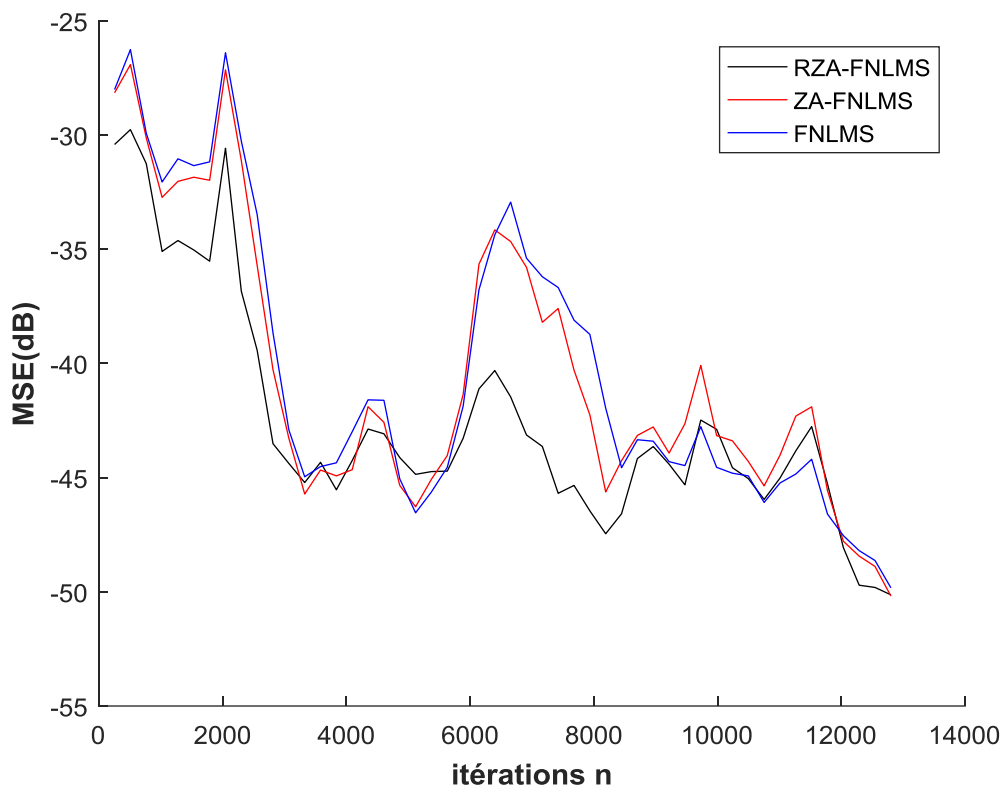


Figure3. 29: signal d'entrée non-stationnaire (Parole), RI synthétique (512) avec  $L=256$ ,  $\psi=10$ ,  $\mu=1$ ,  $SNR=50dB$ .



Maintenant, si nous augmentons la taille de filtre et avec RI parcimonieuse et un SNR fixe à 50dB, nous avons obtenu les figures ci-dessous :

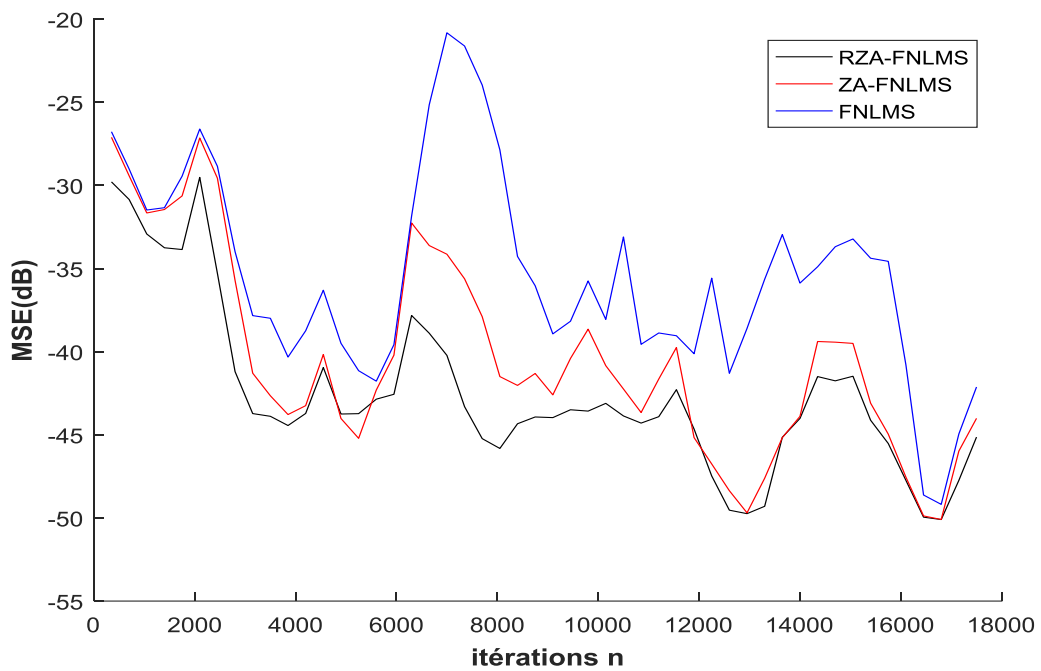


Figure3. 30 signal d'entrée non-stationnaire (Parole), RI synthétique (512) avec  $L=350$ ,  $\psi=10$ ,  $\mu=1$ , SNR=50dB.

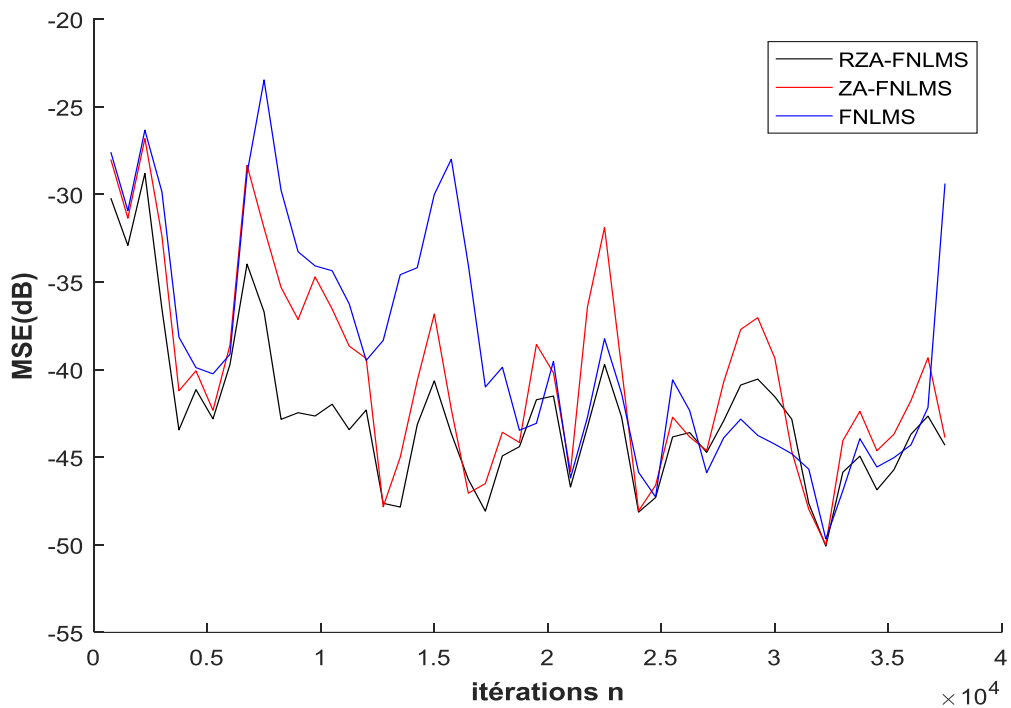


Figure3. 31: signal d'entrée non-stationnaire (Parole), RI synthétique avec  $L=750$ ,  $\psi=10$ ,  $\mu=1$ , SNR=50dB.

Les résultats dans les figures (3.30) et (3.31) nous donnent une remarque sur l'indépendance de la vitesse de convergence d'algorithme RZA-FNLMS de la taille de filtre.

### 3.8 L'algorithme EM-RLS et RLS

Dans cette section nous allons présenter les résultats de simulation que nous avons réalisées avec l'entrée USASI et la réponse impulsionnelle synthétique parcimonieuse et dispersive, cette simulation utilise deux valeurs de SNR pour voir les performances des algorithmes.

Les figures 3.32 et 3.33 nous montrent que l'algorithme EM-RLS a une vitesse de convergence rapide par rapport à l'algorithme RLS dans le cas où le SNR est faible. Dans la suite nous allons varier le degré de parcimonie avec un SNR fixe à 25 dB.

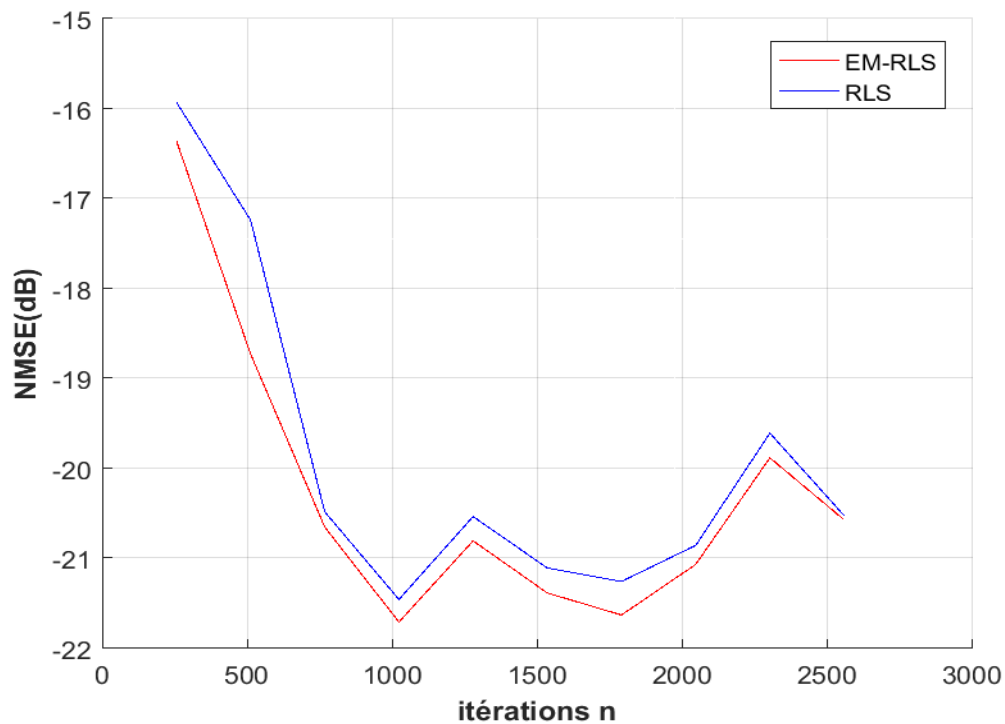


Figure3. 32: Signal d'entrée stationnaire (USASI), RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\lambda=0.995$ ,  $\delta=0.33$ ,  $SNR=25dB$ .

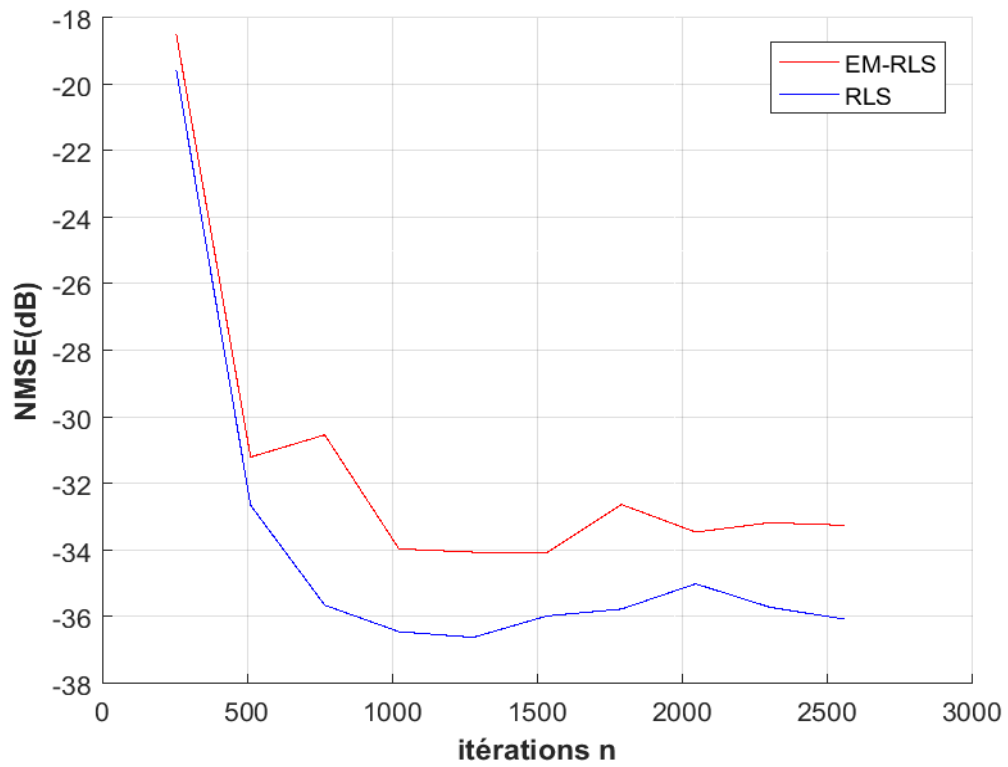


Figure 3.33 : Signal d'entrée stationnaire (USASI), RI synthétique avec  $L=256$ ,  $\psi=10$ ,  $\lambda=0.995$ ,  $\delta=0.33$ ,  $SNR=40dB$ .

D'après les résultats obtenus dans les figures 3.32 et 3.34 nous remarquons que l'algorithme de EM-RLS converge rapidement avec les RIP et les RID, mais les algorithmes EM-RLS et RLS ont même erreur en régime permanent avec les RID.

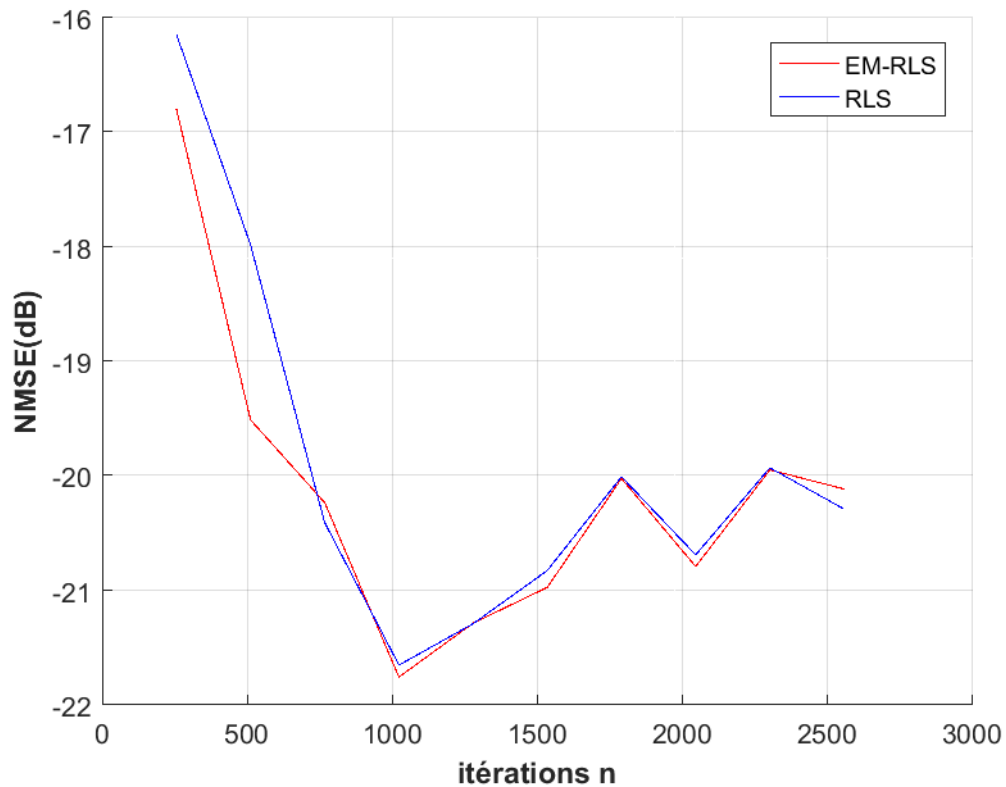


Figure3. 34: Signal d'entrée stationnaire (USASI), RI synthétique avec  $L=256$ ,  $\psi=60$ ,  $\lambda=0.995$ ,  $\delta=0.33$ ,  $SNR=25dB$ .

### 3.9 Conclusion

Dans ce chapitre nous avons comparé les performances des filtres adaptatifs de deux catégories d'algorithmes avec des signaux d'entrées stationnaires et non-stationnaires filtrés par des RI synthétique de différents degrés de parcimonie et de RI réels dans plusieurs environnements (habitable d'une voiture et salle audioconférence).

Dans la première catégorie, nous avons présenté les résultats de simulations des algorithmes moindres carrés qui nous permettent de conclure en général que la vitesse de convergence des algorithmes RZA-(F)NLMS et ZA-(F)NLMS est rapide que (F)NLMS dans le cas des RI parcimonieuses.

Dans la deuxième catégorie, les simulations des algorithmes de moindres carrés récursifs et sa version améliorée, nous donnons une conclusion sur le bon comportement de l'algorithme de EM-RLS que celle de RLS.

## Conclusion générale

---

Dans une première partie, nous avons présenté les algorithmes de base types NLMS et RLS et leurs versions parcimonieuses : ZA-NLMS, RZA-NLMS et EM-RLS. Nous avons aussi décrit l'algorithme à convergence rapide Fast NLMS et nous avons proposé d'étendre les principes de parcimonie du zero-attracting et reweighted- zero-attracting à cet algorithme FNLMS.

Par la suite, nous avons simulé ces algorithmes dans le contexte de l'annulation d'écho acoustique avec des réponses impulsionnelles synthétiques et réelles avec différents degrés de parcimonie et des signaux d'entrées stationnaires (USASI) ou non stationnaire (Parole) avec différents rapports SNR en entrée.

Les algorithmes ZA et RZA-NLMS montrent une supériorité en vitesse de convergence par rapport au NLMS pour les RI parcimonieuses et différents rapports SNR. Pour les RI dispersives, la convergence est presque identique pour les trois algorithmes, et erreur finale du NLMS est même meilleure. Ce comportement a été observé aussi avec les signaux de parole en entrée.

Les algorithmes ZA et RZA FNLMS donnent une vitesse de convergence meilleure par rapport au FNLMS pour des RI parcimonieuses pour différentes tailles du filtre et différents SNR. L'amélioration est nettement visible dans le cas d'un signal de parole en entrée.

La comparaison de l'algorithme RLS et sa version parcimonieuse EM-RLS, montre que ce dernier améliore la vitesse de convergence pour des SNR entre 15 et 25 dB pour les RI parcimonieuses et dispersives.

Enfin comme perspective, nous proposons d'étendre les principes de parcimonie considérés dans ce projet aux algorithmes moindres carrés rapides type FTF.

# Bibliographie

---

- [1] A. TIADJANI, "Study of sparse adaptive algorithm for identification of acoustic impulse response," Blida, 2016.
- [2] S. Haykin, *Adaptive Filter Theory*, 3, Ed., New Jersey: Prentice Hall , 1996.
- [3] D. e. K. Monolakis, *Statistical and adaptive Signal Processing*, Artech House , 2005.
- [4] Turan,C, "SPARSE ADAPTIVE FILTERING TECHNIQUES FOR ACOUSTIC ECHO CANCELLATION," Mevlana University,2016.
- [5] M. AREZKI, "Algorithmes adaptatifs évolués de filtrage," thèse de doctorat en électronique, ENP, Alger, 2011.
- [6] M. S. Salman, N. Jahromi, A. Hocanin and O. Kuker, "A zero-attracting Variable step size LMS algorithm for sparse system identification," IX International symposium on télécommunication, Sarajevo, 2012.
- [7] C. Turan and M. Salman, "A sparse function controlled variable step size LMS algorithm for system identification," in *IEEE 22nd signal processing and communication application conference*, Trabzon,Turkey, 2014.
- [8] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J.Royal*, vol. 58, pp. 267-288, 1996.
- [9] Y. L. S. e. S. J. Avishy, "Compressed Sensing and sparse Filtering," *Springer-Verlag*, p. 501, 2014.
- [10] G. Gui and F. Adachi, "Improved least mean square algorithm with application to adaptive sparse channel estimation," *EURASIP*, vol. 1, pp. 1-18, 2013.
- [11] Y. Chen, Y. Gu and O. Alfred, "Sparse LMS for system identification," *Acoustics, Speech and Signal Processing*, p. 4, 2009.

- [12] H. Li, C. Shen and Q. Shi, "Real-Time visual tracking using compressive sensing," in *Proceeding of the IEEE conference on computer vision and pattern recognition*, 2011.
- [13] G. Gui, W. Peng and F. Adachi, "improved adaptive sparse channel estimation based on the least mean square algorithm," in *IEEE wireless communication and networking conference (WCNC)*, Shanghai,China, 2013.
- [14] F. wen, "System identification using reweighted zero attracting least absolute deviation algorithms," pp. 1-5, 2012.
- [15] A. BENALLAL and M. AREZKI, "A fast convergence normalized least mean square type algorithm for adaptive filtering," *International journal of adaptive control and signal processing* , p. 8, 2013.
- [16] A. Benallal and A. Benkrid, "A simplified FTF algorithm for adaptive filtering," *Signal processing* , pp. 904-917, 2007.
- [17] M. Arezki, A. Benallal, P. Meyrueis and D. Berkani , "A new algorithm with low complexity for adaptive filtering," *IAENG journal*, pp. 205-211, 2010 .
- [18] J. Cioffi and T. Kailath, "Fast recursive transversal filter for adaptive filtering," *IEEE transactions on acoustics Speech and signal processing* , pp. 304-337, 1984.
- [19] Rupp M. A family of adaptive algorithms with decorrelating properties. *IEEE Transactions Signal Processing*, 1998.
- [20] Nicholas,K;Gersimos,M;Behtash,B;Vahid,T," Adaptive algorithms for sparse system identification", ELSEVIER, signal processing,1910-1919,2011.
- [21] Jacob,B;Silviu,C,"Sparse adaptive filters for echo cancellation",Morgan &Claypool,2010.
- [22] Khong,A.W.H; Naylor,P.A,"Efficient use of sparse adaptive filters", Proc.Fortieth Asilomar Conf. Signal, Systems and computers, 1375-1379,2006.