
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 البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Filière télécommunication
Spécialité Système de télécommunication

MESBAH Samia

&

ABIDAT yousra

Algorithme NLMS optimisé avec structure de blanchiment

Proposé par : BENALLAL Ahmed.

Année Universitaire 2019-2020

ملخص :

يأتي هذا العمل في إطار إلغاء الصدى الصوتي من خلال الأساليب التكيفية. لقد درسنا ثم برمجنا في بيئة Matlab إصدارًا جديدًا من خوارزمية OP-NLMS-BL استنادًا على تقنية التبييض المتلاشية انطلاقًا من خوارزمية OP-NLMS. الغرض من المحاكاة هو إجراء مقارنة الخوارزميات وتحسينها من حيث سرعة التقارب والخطأ النهائي. أظهرت النتائج المتحصل عليها أن خوارزمية OP-NLMS-BL تقدم تحسینًا شاملاً من حيث سرعة التقارب والخطأ النهائي مقارنةً بخوارزمية OP-NLMS الكلاسيكية، مما يثبت أن OP-NLMS-BL هو مرشح مثالي لتطوير الاتصالات اللاسلكية المستقبلية.

الكلمات الجوهرية: إلغاء الصدى الصوتي ; التكيف OP-NLMS-BL ; OP-NLMS ;

Résumé : ce travail porte une étude sur le domaine d'annulation d'écho acoustique par des approches adaptatives. nous avons étudié puis programmé sous l'environnement matlab une nouvelle version d'algorithme OP-NLMS-BL fondue à partir de l'algorithme OP-NLMS en s'appuyant sur la technique de blanchiment. Le but des simulations effectuées est la comparaison et l'amélioration des algorithmes vis-à-vis la vitesse de convergence et de l'erreur finale.

Les résultats obtenus montrent que l'algorithme OP-NLMS-BL offre une amélioration globale en terme de vitesse de convergence par rapport à l'algorithme OP-NLMS classique, prouvant que l'OP-NLMS-BL est un candidat idéal pour le développement des communications sans fils future.

Mots clés : annulation d'écho acoustique ; adaptatif; OP-NLMS-BL ; OP-NLMS.

Abstract : This work is within the framework of acoustic echo cancellation by means of adaptative approaches . we have studied and then programmed under a MATLAB environment a new version of the OP-NLMS-BL algorithm based on the fade whitening technique based on the OP-NLMS algorithm. the purpose of the simulations performe dis to compare and improve algorithms in terms of convergence speed and final error. the results obtained show that the OP-NLMS-BL algorithm offers an overall improvement in terms of convergence speed compared of the classical OP-NLMS algorithm, proving that the OP-NLMS-BL is an ideal condidate for development of future wireless communications.

Keywords : acoustic echo cancellation ; adaptative ; OP-NLMS-BL ; OP-NLMS.

Dédicaces

Que ce travail témoigne de mes respects :

A mes parents Djelloul et Hdjila :

Grace à leurs tendres encouragements et leurs grands sacrifices, ils ont pu créer le climat affectueux et propice à la poursuite de mes études. Aucune dédicace ne pourrait exprimer mon respect, ma considération et mes profonds sentiments envers eux. Je prie Allah de les bénir, de veiller sur eux, en espérant qu'ils seront toujours fières de moi.

A ma chère sœur Amina, Ahlem ,Imene et meriem et mes chère frères, Mohamed, Salim et merouan En témoignage de l'attachement, de l'amour et de l'affection que je porte pour vous.

Je vous dédie ce travail avec tous mes vœux de bonheur, de santé et réussite.

A tous les membres de ma famille petits et grands .

Veillez trouver dans ce modeste travail l'expression de mon affection.

A mon fiancé AIT SAADI Younes mon soutien moral et ma source de joie et de bonheur.

A ma belle famille Je vous remercie tout particulièrement pour votre soutien. Puissiez-vous trouver dans ce travail le témoin de mon affection et estime.

A ma meilleure amie yousra Je te remercie pour ton soutien inconditionnel durant toutes ces longues années d'études. Ton amour et ton affection remplissent mes jours de bonheur.

A mes amies a ma chère amie Nora, je te remercie pour ton amitié chère à mon cœur, et je te souhaite tout le bonheur du monde. Toute mon affection pour ton admirable famille, que je remercie beaucoup. A tous mes amis, chahinez , mounia et chahinez .

SAMIA

Dédicaces

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

Je dédie ce modeste travail :

À mes parents WIDED et MOUHAMED qui depuis mon plus jeune âge ont toujours fait leur maximum, en consacrant temps et argent, pour m'éveiller et m'encourager dans mes passions.

C'est grâce à vous et pour vous que j'ai fait mon mémoire. Aucun mot sur cette page ne saurait exprimer ce que je vous dois, ni combien je vous aime. Qu'Allah vous bénisse, vous assiste, vous vienne en aide

A mes chères sœurs plus particulièrement HOUDA pour leurs encouragements permanents,

et leur soutien moral.

A mon cher frère MALIK pour leur appui et leur encouragement

A chère grand-mère, mes oncles, mes tantes et tous les membres de la Famille ABIDAT et

TOBNI

A mon amie avant qu'elle être mon binôme SAMIA, je dis merci d'être avec moi et

d'avoir partagé ce travail

En témoignage de mon amitié sincère ;A tous mes amis, plus particulièrement :

Selma, Kenza, Rachida, Meriem, Randa pour leur appui et leurs encouragements.

YOUSRA

Remerciements

Louange à Allah, seigneur de l'univers, pour la volonté, la santé, le courage et la patience pour accomplir ce modeste travail.

Nous tenons ainsi, à exprimer ici notre respect et toute notre reconnaissance à notre encadreur Monsieur : BENALLAL Ahmed, pour sa bienveillance, sa gentillesse, ses encouragements et ses conseils.

Nous remercions vivement les membres du jury pour avoir accepté d'évaluer ce modeste mémoire.

En fin, nous tenons à remercier tous ceux qui ont contribué à ce travail de près ou de loin.

Table des matières :

Introduction générale.....	1
1. CHAPITRE 1 : Echo acoustique et filtrage adaptatif	3
1.1 Introduction :.....	3
1.2 L'écho acoustique :.....	3
1.3 Le canal acoustique et La réponse impulsionnelle acoustique (RIA) :	4
1.4 Annulation adaptatif de l'écho acoustique :	5
1.4.1 Filtrage de Wiener :.....	6
1.4.2 Filtrage adaptatif :	8
1.4.3 Les algorithmes adaptatifs :	9
1.4.4 les critères de comparaison des algorithmes adaptatifs :.....	16
1.5 Conclusion :	17
2. CHAPITRE 2 : Algorithmes NLMS Optimisés (OP-NLMS).....	18
2.1 Introduction :.....	18
2.2 L'algorithme OP-NLMS :	18
2.2.1 L'étude de convergence de l'algorithme NLMS :	19
2.2.2 Dérivation de l'algorithme OP-NLMS :	22
2.3 L'algorithme OPNLMS récursif :	23
2.4 Complexité des calculs :	25
2.5 Conclusion :	25
3. CHAPITRE 3 : Algorithme OPNLMS avec blanchiment	26
3.1 Introduction :.....	26
3.2 Principe du blanchiment :	26
3.3 Résumé de méthodes existantes:	27
3.4 La prédiction linéaire :.....	35
3.5 Algorithme OPNLMS-R avec blanchiment :.....	39
3.6 Conclusion :	44
4. CHAPITRE 4 : résultats de simulation	45
4.1 Introduction :.....	45
4.2 Critère de performance MSE :.....	45
4.3 Description des signaux et des paramètres :	45
4.4 Simulations :.....	49
4.5 Conclusion :	59

Conclusion générale61

Listes des figures :

Figure 1-1 : communication bidirectionnelle entre deux salles	3
Figure 1-2 : Exemple d'une réponse impulsionnelle dans une salle de téléconférence ($F_e=16\text{kHz}$) [2].	5
Figure 1-3: système de l'annulation d'écho	5
Figure 1-4: Identification par filtrage adaptatif.....	8
Figure 3-1: Pré-blanchiment sur le chemin d'écho.	27
Figure 3-2: Pré-blanchiment nécessitant la recopie de l'annuleur.	27
Figure 3-3: Blanchiment des signaux de l'algorithme.	28
Figure 3-4: structure à base d'algorithmes de signe.....	29
Figure 3-5: Filtrage adaptatif en sous bandes.....	31
Figure 3-6: Filtrage adaptatif en treillis.....	32
Figure 3-7: Structure treillis simplifiée.	33
Figure 3-8: Prédiction linéaire.....	36
Figure 4-1 : signal usasi.....	46
Figure 4-2: signal BBG1AR20.....	46
Figure 4-3: signal parole SPMX1.....	46
Figure 4-4: CAR impulse response ($L=256$).....	47
Figure 4-5: CAR impulse response ($L=512$).....	47
Figure 4-6: USASI, SNR=15dB, $L=256$ et $P=1$	50
Figure 4-7: USASI, SNR=30dB, $L=256$ et $P=1$	50
Figure 4-8: USASI, SNR=50dB, $L=256$ et $P=1$	50
Figure 4-9: USASI, SNR=15dB, $L=512$ et $P=1$	50
Figure 4-10: BBG1AR20, SNR=15dB, $L=256$ et $P=1$	51
Figure 4-11: BBG1AR20, SNR=30dB, $L=256, P=1$	51
Figure 4-12: BBG1AR20, SNR=15dB, $L=512$ et $P=1$	51
Figure 4-13: BBG1AR20, SNR=30dB, $L=512$ et	51
Figure 4-14: USASI, SNR=15dB, $L=256$ et $P=1,2$ et 10	52
Figure 4-15: USASI, SNR=30dB, $L=256$ et $P=1, 2$ et 10.....	52
Figure 4-16: USASI, SNR=15dB, $L=512$ et $P=1,2$ et 10.....	53
Figure 4-17: BBG1AR20, SNR=15dB, $L=256$ et $P=1,2$ et 7.....	53
Figure 4-18: USASI, SNR=15dB, $L=256$ et $P=1$ récursif.....	54
Figure 4-19: USASI, SNR=30dB, $L=256$ et $P=1$ récursif.....	54
Figure 4-20: USASI, SNR=50dB, $L=256$ et $P=1$ récursif.....	54
Figure 4-21: USASI, SNR=15dB, $L=512$ et $P=1$ récursif.....	54
Figure 4-22: BBG1AR20, SNR=15dB, $L=256$ et $P=1$ récursif	55
Figure 4-23: BBG1AR20, SNR=30dB, $L=256$ et $P=1$ récursif.....	55
Figure 4-24: BBG1AR20, SNR=50dB, $L=256$ et $P=1$ récursif.....	55
Figure 4-25: BBG1AR20, SNR=30dB, $L=512$ et $P=1$ récursif.....	55
Figure 4-26: USASI, SNR=15dB, $L=256$ et $P=1, 2$ récursifs.	56
Figure 4-27: BBG1AR20, SNR=15dB, $L=512$ et $P=1, 2$ récursifs.	56
Figure 4-28: USASI, SNR=30dB, $L=512$ et $P=1,2$ récursifs.	56
Figure 4-29: BBG1AR20, SNR=30dB, $L=256$ et $P=1,2$ récursifs.	56
Figure 4-30: Parole SPMX1, SNR=15dB, $L=256$, $P=1$, $\lambda=0.9986$	58
Figure 4-31: Parole SPMX1, SNR=15dB, $L=256$, $P=1$, $\lambda=0.9986$	58

Figure 4-32: Parole SPMX1, SNR=50dB, L=256, P=1 R, $\lambda=0.9986$	58
Figure 4-33: Parole SPMX1, SNR=15dB, L=512, P=1 R, $\lambda=0.9986$	59
Figure 4-34: Parole SPMX1, SNR=30dB, L=512, P=1 R, $\lambda=0.9986$	59
Figure 4-35: Parole SPMX1, SNR=50dB, L=512, P=1 R, $\lambda=0.9986$	59

Listes des tableaux :

Tableau 1-1 :Résumé de l'algorithme LMS.	11
Tableau 1-2 :Résumé de L'algorithme NLMS.	11
Tableau 1-3: Résumé de L'algorithme RLS.	14
Tableau 1-4: Résumé de L'algorithme FRLS	16
Tableau 1-5: Les critères de comparaison des algorithmes adaptatifs.	17
Tableau 2-1: Résumé de l'algorithme OP-NLMS.	23
Tableau 2-2: Résumé de l'algorithme OP-NLMS-REC.	24
Tableau 2-3: complexité de calculs.	25
Tableau 3-1: L'algorithme OPNLMS-R avec blanchiment.	41

Listes des acronymes et abréviations :

HP	: hautparleur.
MIC	: microphone.
RI	: R éponse I mpulsionnelle.
RIF	: R éponse I mpulsionnelle F inie.
RIA	: R éponse I mpulsionnelle A coustique.
EQM	: E rreur Q uadratique M oyenne.
LMS	: L east M ean S quare.
NLMS	: N ormalized L east M ean S quares.
LS	: L east S quares.
RLS	: R ecursive L east S quares.
FRLS	: F ast R ecursive L east S quares.
OP-NLMS	: o ptimized N ormalized L east M ean S quares.
OP-NLMS-REC	: o ptimized N ormalized L east M ean S quares r ecursive.
OP-NLMS-BL	: o ptimized N ormalized L east M ean S quares b lanchit.
LPC	: L inear P rediction C oefficient .

Notations :

- $[.]^T$: Opération de transposition.
- $[.]^{-1}$: Opération de matrice inverse.
- $|\cdot|$: Opération valeur d'absolue.
- $E(\cdot)$: Espérance mathématique.
- $trace(\cdot)$: Trace d'une matrice.
- $\|\cdot\|$: Norme d'un vecteur.
- $\|\cdot\|_1$: La norme l_1 .
- $\|\cdot\|_2$: La norme l_2 .
- ∇ : Vecteur de gradient.
- $\partial(\cdot)/\partial(\cdot)$: Dérivé partiel.

Introduction générale

Introduction générale :

Que ce soit la téléphonie, la télévision ou l'Internet, il est évident que nous sommes constamment entourés par des systèmes de communications. Ces derniers nous poussent d'ailleurs d'augmenter continuellement les capacités des systèmes de transmissions. De ce fait ces systèmes devront avoir une qualité satisfaisante pour permettre aux personnes de s'échanger des informations sans déplacement.

Toutefois, la qualité de la communication est souvent fortement dégradée par le phénomène d'écho acoustique, qui s'avère intolérable lors d'une conversation conviviale. Pour cette raison Un traitement appuyé sur le filtrage adaptatif doit être impérativement mis en œuvre.

Le principal intérêt du filtrage adaptatif consiste à éliminer le bruit dont les caractéristiques évoluent dans le temps par des différents algorithmes adaptatifs, ces derniers se divisent en deux familles ,la famille des moindres carrés et la famille des gradients stochastiques.

Le but de notre projet de fin d'étude est de mettre l'accent sur un nouvel algorithme adaptatif NLMS optimisé avec une structure de blanchiment et de le comparer avec l'algorithme NLMS optimisé classique et essayer de mettre en évidence ses critères de performances.

Ce mémoire est constitué de quatre chapitres :

Le premier chapitre est introductif, il décrit brièvement l'écho acoustique et le canal acoustique ainsi que sa réponse impulsionnelle, en passant par l'annulation adaptatif de l'écho acoustique et par le filtrage de Wiener. Dans un second lieu nous avons défini les algorithmes adaptatifs en évoquant ses deux familles, la famille des gradients stochastiques LMS ensuite la

famille des moindres carrés LS. Finalement Les critères de comparaisons des algorithmes adaptatifs sont cités.

Dans le second chapitre nous avons présenté l’algorithme NLMS optimisé, puis nous avons aussi étudié la convergence de l’algorithme NLMS plus la dérivation de l’algorithme NLMS optimisé, au finale nous avons expliqué l’algorithme NLMS optimisé récursif ,la complexité des calculs des algorithmes traités précédemment est également exposé.

Dans le troisième chapitre, nous avons incorporé dans un premier temps le principe du blanchiment et un résumé des méthodes existantes. En deuxième lieu, la prédiction linéaire et en terminant avec l’algorithme NLMS optimisé récursif avec blanchiment.

Dans le dernier chapitre, nous avons défini les critères de performance en apportant une description sur signaux et les paramètres utilisés dans la simulation .

Des exécutions sur MATLAB sont accomplies pour comparer les performances de l’algorithme proposé et l’algorithme classique , en employant :

- La prédiction lineaire par matlab (LPC).
- LPC matlab avec un prédicteur récursif a premier ordre.
- LPC matlab avec un prédicteur récursif a second ordre.

A la fin de ce manuscrit, une conclusion générale récapitule les apports essentiels de notre travail.

Chapitre 1

1. CHAPITRE 1 : Echo acoustique et filtrage adaptatif

1.1 Introduction :

La source de l'écho acoustique dans le contexte "mains libres" s'avère intolérable pour la téléphonie, pour cela un traitement déterminé est appliqué afin de fournir un environnement sans écho. Au cours de ce chapitre nous allons aborder la source de l'écho en évoquant les algorithmes de filtrage adaptatif qui ont déjà été développés pour sa suppression.

1.2 L'écho acoustique :

L'écho est un phénomène acoustique engendré par le couplage entre le hautparleur et le microphone lors d'une communication bidirectionnelle entre deux salles (figure 1.1), le signal transmis par la salle distante est réémis par la même salle, lorsque le retard de transmission entre les deux salles est trop important. Par conséquent les locuteurs présents dans cette salle vont réentendre leur propre voix. L'écho acoustique devient très gênant à partir d'un délai de transmission global de 30ms. Pour cela un système d'atténuation d'écho doit être inséré dans la chaîne de communication [1].

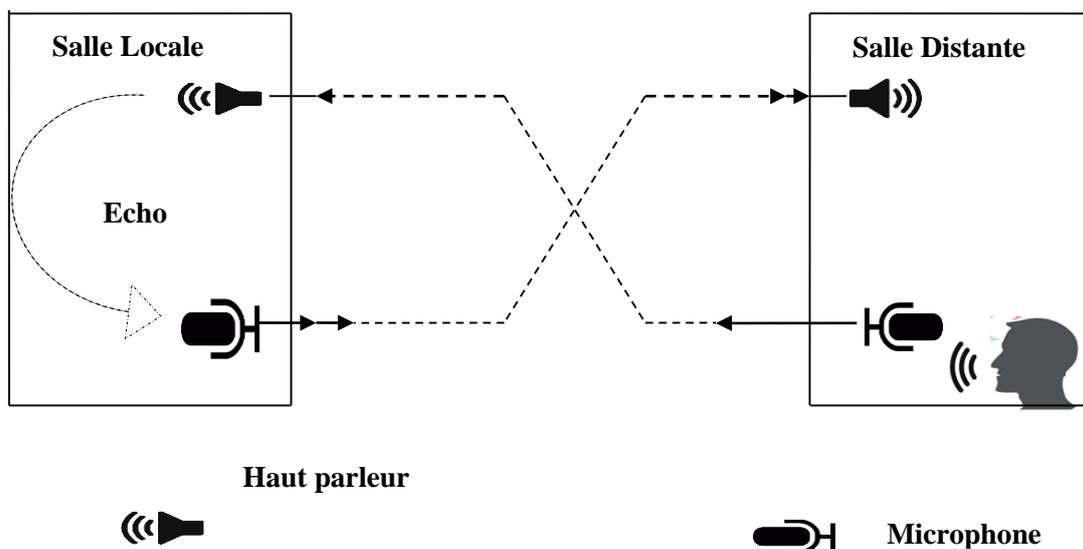


Figure 1-1 : communication bidirectionnelle entre deux salles .

1.3 Le canal acoustique et La réponse impulsionnelle acoustique

(RIA) :

Le canal acoustique est par définition, la transformation du signal original transmis par une source émettrice (ex. haut parleur HP, locuteur) et reçu par un récepteur(ex. microphone MIC ou oreille humaine)[3].

Les éléments essentielles liés au canal acoustique sont la position de l'émetteur et du récepteur ainsi que leurs diagrammes et propriétés acoustiques du milieu(les murs, les objets de la salle).D'après l'approximation de l'acoustique linéaire, le canal représente un filtre linéaire de durée finie RIF, sa sortie est donnée par la somme de convolution suivante :

$$y_n = \mathbf{x}_n^T \mathbf{h}_n \quad (1.1)$$

Avec :

\mathbf{w}_n : Réponse impulsionnelle (RI) du canal.

\mathbf{x}_n : Signal d'entrée.

La RIA est la quantité mesurée entre le haut parleur et le microphone, elle se présente sous forme d'une onde directe et d'une succession d'ondes réfléchies, diffractées ou encore diffusées par les parois de la salle, en pratique la taille de la RIA dépend des dimensions physiques du milieu acoustique (salle, habitacle d'une voiture), elle varie d'une centaine de points, pour les petites salles et à quelques milliers de points pour les grandes salles (figure 1.2)[3]. Une autre contrainte est que les mouvements de personnes ou d'objets dans la salle rendent la RIA variable ou non stationnaire. Ce qui explique l'utilisation du filtrage adaptatif lors de l'identification ou l'annulation de l'écho acoustique de couplage [4].

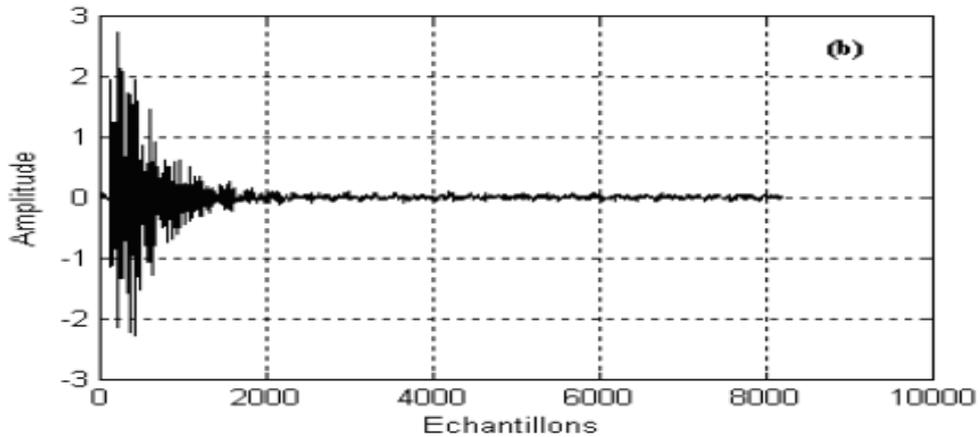


Figure 1-2 : Exemple d'une réponse impulsionnelle dans une salle de téléconférence (Fe=16khz) [2].

1.4 Annulation adaptatif de l'écho acoustique :

Dans une communication de type "main libre" le couplage acoustique entre le haut parleur et le microphone occasionne un écho gênant chez le locuteur, il est donc imposant pour ce système de disposer une solution afin d'éliminer l'écho présent dans le signal émis. Le schéma proposé dans la figure (1.3) illustre le traitement établi pour annuler l'écho acoustique [5].

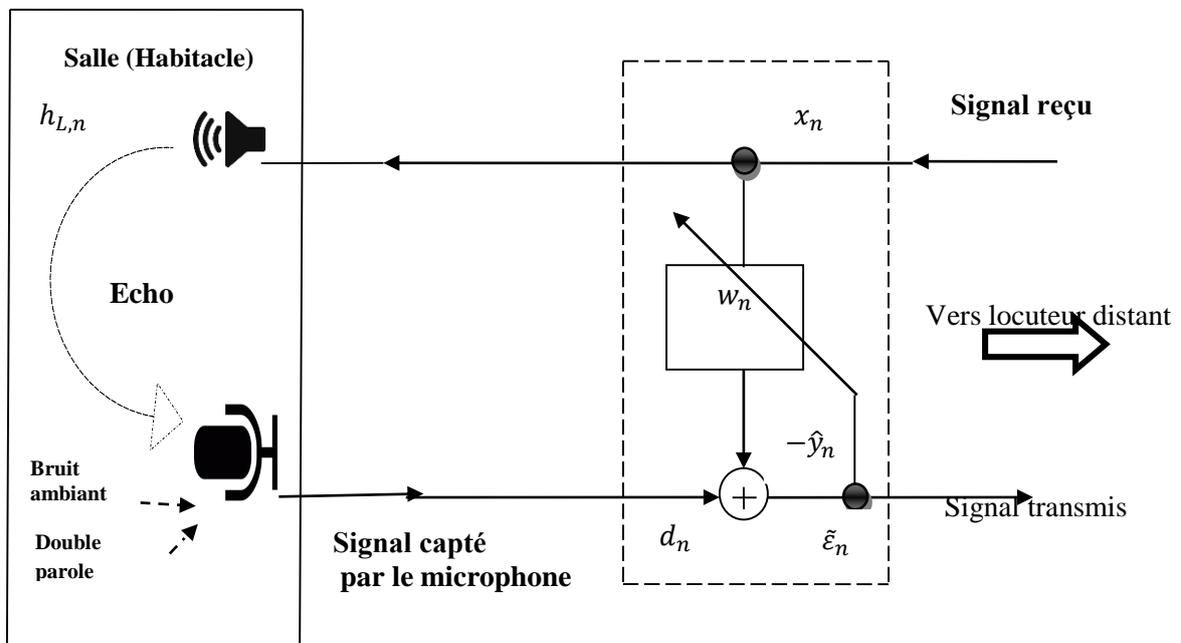


Figure 1-3: système de l'annulation d'écho .

Le signal x_n est transmis par le haut parleur ensuite réfléchi sur les parois et les objets de la salle, le microphone capte le signal résultant y_n (signal d'écho). Ce signal est simulé par un filtrage linéaire du signal transmis x_n . Le chemin acoustique réel entre le haut parleur et le microphone est caractérisé par une réponse impulsionnelle h .

Le signal d_n reçu par le microphone présente la somme du signal d'écho y_n , du bruit ambiant v_n et le signal utile s_n (signal double parole). Dans ce projet, nous allons supposer ce signal nul.

$$d_n = y_n + s_n + v_n \quad (1.2)$$

L'annulation d'écho a pour but l'estimation du signal y_n en déterminant le chemin acoustique, un signal exemple est délivré après la soustraction de l'estimation du signal d_n [2].

1.4.1 Filtrage de Wiener :

Dans de nombreuses applications, les signaux temporels sont entachés d'une interférence ou d'un bruit non désiré, ce qui nécessite une solution optimale qui permet de supprimer ou réduire ces composantes perturbatrices. Afin de trouver cette solution, le filtre de Wiener minimise la moyenne statistique du carré de l'erreur (EQM) entre l'information désirée et la sortie du filtre. Cette dernière a une réponse \hat{y}_n très proche de la réponse désirée d_n , [5].

La sortie estimée \hat{y}_n s'écrit sous la forme:

$$\hat{y}_n = \mathbf{w}_{n-1}^T \mathbf{x}_n \quad (1.3)$$

où :

\mathbf{x}_n : Vecteur signal d'entrée.

\mathbf{w}_n : Vecteur des coefficients de filtre.

Le signal d'erreur est défini par :

$$\tilde{\varepsilon}_n = d_n - \hat{y}_n \quad (1.4)$$

Le rôle de l'algorithme de Wiener est d'estimer le système représenté par la réponse impulsionnelle, l'estimation du vecteur \mathbf{w}_n est réalisée en minimisant la fonction de coût $J_n(\mathbf{w})$ qui dépend de l'erreur notée dans l'équation (1.4). L'objectif principal est de réduire l'erreur du filtre optimal, les coefficients de ce dernier doivent être très proches que le signal désiré.

$$J_n(\mathbf{w}) = E\{(d_n - \mathbf{w}_n^T \mathbf{x}_n)^2\} \quad (1.5)$$

Le vecteur optimum \mathbf{w}_{opt} annule le gradient du critère, il est donné par l'expression suivante :

$$\mathbf{w}_{opt} = \arg \min J_n(\mathbf{w}) \quad (1.6)$$

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

$$\nabla_{\mathbf{w}} J_n(\mathbf{w}) = \frac{\partial J_n}{\partial \mathbf{w}} = 0 \quad (1.7)$$

$$2\mathbf{R}_{xx}\mathbf{w}_n - 2\mathbf{p}_{xd} = 0 \quad (1.8)$$

Sachant que :

$$\mathbf{R}_{xx} = E(\mathbf{x}_n \mathbf{x}_n^T) \quad (1.9)$$

\mathbf{R}_{xx} : Matrice d'autocorrélation du vecteur \mathbf{x}_n .

$\mathbf{p}_{xd} = E\{\mathbf{x}_n d_n\}$: Vecteur d'intercorrélacion.

Si la matrice d'auto corrélacion est inversible alors le filtre optimal est donné par l'expression suivante :

$$\mathbf{w}_{opt} = \mathbf{R}_{xx}^{-1} \mathbf{p}_{xd} \quad (1.10)$$

En pratique, les propriétés statistiques R_{xx} et p_{xd} sont soit inconnues, soit non stationnaires, nous allons par la suite exposer des méthodes d'estimation adaptatives pour atteindre la solution de Wiener (1.10) [2].

1.4.2 Filtrage adaptatif :

Les filtres classiques rencontrent des difficultés pour identifier les réponses impulsionnelles variables, ce qui provoque un problème majeur dans l'annulation d'écho, pour atteindre l'objectif souhaité, de nouveaux filtres sont proposés appelés « filtres adaptatifs », ces derniers peuvent s'harmoniser aux variations des réponses impulsionnelles du système, afin d'annuler efficacement l'écho acoustique présent (figure 1.4) [6].

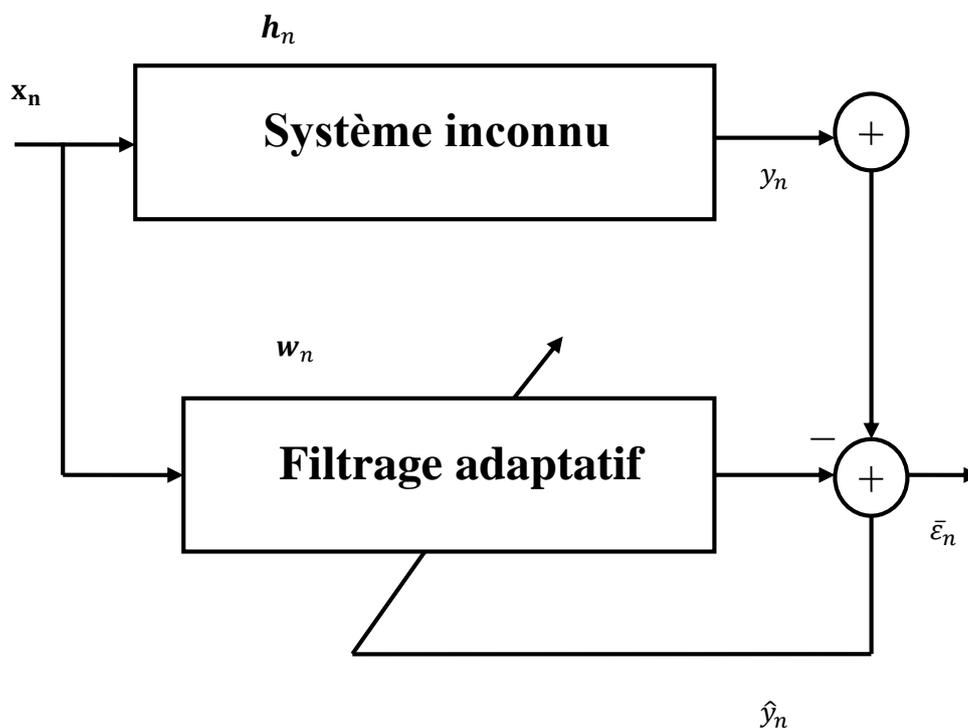


Figure 1-4: Identification par filtrage adaptatif.

x_n : Signal de l'entrée

Avec :

$$\mathbf{x}_n = [x_n, x_{n-1}, \dots, x_{n-L+1}]^T \quad (1.11)$$

y_n : signal de sortie du filtre.

d_n : Echo ou signal désiré.

Sachant que :

$$d_n = \mathbf{x}_n^T \mathbf{h}_n + v_n \quad (1.12)$$

$$d_n = y_n + v_n \quad (1.13)$$

v_n : Signal de bruit additif de moyen nul.

$\tilde{\varepsilon}_n$: Erreur obtenue par la méthode de filtrage adaptatif.

\mathbf{w}_n : réponse impulsionnelle du système (haut parleur – microphone).

$$\mathbf{w}_n = [w_{1,n}, w_{2,n}, \dots, w_{L,n}]^T \quad (1.14)$$

1.4.3 Les algorithmes adaptatifs :

Dans cette partie nous avons énoncé les algorithmes largement utilisés en filtrage adaptatif, soit la famille de gradient stochastique LMS, la famille de moindres carrées récursifs RLS.

a. La famille LMS :

L'algorithme LMS (Least mean squares) est considéré comme l'un des algorithmes les plus efficaces pour le calcul des coefficients d'un filtre en raison de sa robustesse et sa simplicité d'estimation, cependant il est impuissant pour délier l'écho acoustique à cause de la variation d'énergie continue dans le signal de la parole [7] .

Le principe du gradient stochastique consiste à remplacer la moyenne statistique dans l'algorithme de gradient déterministe de l'équation (1.16) par sa valeur instantanée.

$$\mathbf{w}_n = \mathbf{w}_{n-1} - \mu [\nabla_w E\{\tilde{\varepsilon}_n^2\}] \quad (1.15)$$

μ : Pas d'adaptation (contrôle la convergence de l'algorithme).

En développant la fonction $\nabla_w E\{\bar{\varepsilon}_n^2\}$:

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu E\{\mathbf{x}_n \bar{\varepsilon}_n\} \quad (1.16)$$

La mise à jour de l'équation impose que le facteur de convergence soit choisi dans une gamme déterminée afin d'assurer la convergence.

$$\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu \mathbf{x}_n \bar{\varepsilon}_n \quad (1.17)$$

La complexité de l'algorithme LMS est "2N" multiplications pour chaque itération, la condition de convergence de cet algorithme est donnée par un pas d'adaptation μ qui satisfait la condition suivante :

$$0 < \mu \leq \frac{1}{\text{tr}(\mathbf{R}_{xx})} = \frac{1}{L\sigma_x^2} \quad (1.18)$$

$\text{tr}(\cdot)$: désigne la somme des éléments de la diagonale de la matrice d'autocorrélation du signal d'entrée \mathbf{R}_{xx} et σ_x^2 désigne la variance du signal d'entrée. L'algorithme LMS se résume dans le tableau suivant :

Initialisation : $\mathbf{w}_0 = \mathbf{0}_L$;

Condition de convergence : $0 < \mu \leq \frac{1}{\text{tr}(\mathbf{R}_{xx})} = \frac{1}{L\sigma_x^2}$;

Variables disponible à l'instant n : \mathbf{w}_{n-1} ;

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

Équation de mise à jour : $\mathbf{w}_n = \mathbf{w}_{n-1} + 2\mu \mathbf{x}_n \bar{\varepsilon}_n$;

Fin

Tableau 1-1 Résumé de l'algorithme LMS.

Pour les signaux non-stationnaires (l'énergie du signal d'entrée varie avec le temps), l'algorithme LMS a du mal à fonctionner correctement puisque le pas d'adaptation μ est constant. Ce dernier doit être bien choisi pour assurer la bonne convergence de l'algorithme LMS.

a.1 Algorithme du gradient stochastique normalisé NLMS :

L'algorithme de gradient stochastique NLMS est une version modifiée de l'algorithme LMS, dont le pas d'adaptation est normalisé par l'énergie du signal d'entrée.

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + c_0} \quad (1.19)$$

c_0 : constante (pour éviter la division par 0 et $0 < \mu < 2$).

L'algorithme NLMS se résume dans le tableau suivant :

Initialisation : $\mathbf{w}_0 = \mathbf{0}_L$;
Condition de convergence : $0 < \mu < 2$;
Variables disponible à l'instant n : \mathbf{w}_{n-1} ;
Erreur de filtrage : $\bar{\varepsilon}_n = d_n - \mathbf{w}_{n-1}^T \mathbf{x}_n$;
Equation de mise à jour : $\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + c_0}$;
Fin

Tableau 1-2 :Résumé de L'algorithme NLMS.

b La famille d'algorithmes des moindres carrés LS:

b.1 Solution des moindres carrés LS :

Dans la méthode des moindres carrés, au lieu de minimiser un critère basé sur l'erreur quadratique moyenne, nous minimisons un critère déterministe basé sur une somme pondérée des carrés d'erreurs depuis l'instant initial donné par l'expression suivante :

$$J_n(\mathbf{w}) = \sum_{i=1}^n \lambda^{n-i} (y_i - \mathbf{w}_{L,n}^T \mathbf{x}_{L,i})^2 \quad (1.20)$$

Où

λ : Facteur de pondération pour oublier le passé lointain compris entre 0 et 1..

Nous admettons que les signaux \mathbf{x}_n et y_n sont nuls avant l'instant initial $n=0$, la solution qui exprime la nullité du gradient de la fonctionnelle $J_n(\mathbf{w})$ est représentée par :

$$\nabla_{\mathbf{w}} J_n(\mathbf{w}) = 0 \Leftrightarrow -2 \left[\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L,i} y_i - \left(\sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L,i} \mathbf{x}_{L,i}^T \right) \mathbf{w}_{L,n} \right] = 0 \quad (1.21)$$

Avec :

$$\mathbf{R}_{L,n} \mathbf{w}_{L,n} = \mathbf{P}_{L,n} \quad (1.22)$$

la solution optimale des moindres carrés est comme suit:

$$\mathbf{w}_{opt} = \mathbf{R}_{L,n}^{-1} \mathbf{P}_{L,n} \quad (1.23)$$

$\mathbf{R}_{L,n}$: Matrice d'autocorrélation à court terme, s'écrit sous la forme :

$$\mathbf{R}_{L,n} = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L,i} \mathbf{x}_{L,i}^T \quad (1.24)$$

$\mathbf{P}_{L,n}$: Vecteur d'inter corrélation à court terme, définît par :

$$\mathbf{P}_{L,n} = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L,i} y_i \quad (1.25)$$

L'algorithme nécessite l'inversion d'une matrice carrée d'ordre L dont le cout de calcul est de l'ordre L^3 d'opérations arithmétiques par itération.

b.2 Algorithme des moindres carrés récursifs (RLS)

Les relations récursives des expressions (1.24) et (1.25) sont données respectivement :

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

$$\mathbf{P}_{L,n} = \lambda \mathbf{P}_{L,n-1} + \mathbf{x}_{L,n} \mathbf{y}_n \quad (1.27)$$

En substituant les quantités ci-dessous :

$$\lambda \mathbf{R}_{L,n-1} = \lambda \mathbf{R}_{L,n} - \mathbf{x}_{L,n} \mathbf{x}_{L,n}^T \quad (1.28)$$

$$\lambda \mathbf{P}_{L,n-1} = \lambda \mathbf{P}_{L,n} - \mathbf{x}_{L,n} \mathbf{y}_n \quad (1.29)$$

De (1.23) on aura :

$$\lambda \mathbf{R}_{L,n-1} \mathbf{w}_{L,n-1} = \lambda \mathbf{P}_{L,n-1} \quad (1.30)$$

La solution équivalente sous une forme récursive est exprimé par:

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \bar{\varepsilon}_{L,n} \mathbf{K}_{L,n} \quad (1.31)$$

$\mathbf{k}_{L,n}$: Vecteur appelé gain de Kalman, noté comme suit :

$$\mathbf{k}_{L,n} = \mathbf{R}_{L,n}^{-1} \mathbf{x}_{L,n} \quad (1.32)$$

$$\mathbf{K}_{L,n} = [\mathbf{K}_{1,n}, \mathbf{K}_{2,n} \dots \dots \dots \mathbf{K}_{L,n}]^T \quad (1.33)$$

L'inverse de la matrice $\mathbf{R}_{L,n}$ est exprimé par l'équation suivante :

$$\mathbf{R}_{L,n}^{-1} = \lambda^{-1} \left[\mathbf{R}_{L,n-1}^{-1} - \frac{\mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n} \mathbf{x}_{L,n}^T \mathbf{R}_{L,n-1}^{-1}}{\lambda + \mathbf{x}_{L,n}^T \mathbf{R}_{L,n-1}^{-1} \mathbf{x}_{L,n}} \right] \quad (1.34)$$

L'algorithme RLS se résume dans le tableau suivant :

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

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

Nouvelles informations : \mathbf{x}_n , \mathbf{y}_n ;

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

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

Adaptation de filtre : $\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \bar{\varepsilon}_{L,n} \mathbf{k}_{L,n}$;

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

Tableau 1-3: Résumé de L'algorithmme RLS.

b.3 L'algorithmmes des moindres carrés récursifs rapides (FRLS) :

Il existe plusieurs versions d'algorithmmes à moindres carrés récursifs rapides (Fast RLS), le Kalman (FK) est la version la plus ancienne. le but de ces algorithmmes est de calculer de gain d'adaptation par des prédicteurs aller et retour.

La variable de vraisemblance et le gain de Kalman dual sont calculés indépendamment de la partie filtrage par un algorithmme FRLS qui emploie une analyse de prédiction linéaire pour le signal d'entréex_n, les solutions récursives sont obtenues après l'implémentation de l'inverse de la matrice de covariance partitionnée suivantes [8] :

$$\mathbf{R}_{L+1,n}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{R}_{L,n-1}^{-1} \end{bmatrix} + \begin{bmatrix} I \\ -\mathbf{a}_{L,n} \end{bmatrix} (I - \mathbf{a}_{L,n}^T) \alpha_{L,n}^{-1} \quad (1.35)$$

$$\mathbf{R}_{L+1,n}^{-1} = \begin{bmatrix} \mathbf{R}_{L,n-1}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{b}_{L,n} \\ I \end{bmatrix} (-\mathbf{b}_{L,n}^T \quad I) \beta_{L,n}^{-1} \quad (1.36)$$

$\mathbf{a}_{L,n}$: Prédicteur d'aller d'ordre L.

$\mathbf{b}_{L,n}$: Prédicteur de retour d'ordre L.

$\alpha_{L,n}$: Variance d'erreur de prédiction d'aller.

$\beta_{L,n}$: Variance d'erreur de prédiction de retour.

En changeant l'indice de temps en (n-1) dans (1.35 et (1.36) puis en multipliant ces expressions par $\lambda^{-1} \mathbf{X}_{L+1,n}$ nous aurons deux formes équivalentes pour le gain de kalman dual étendu d'ordre L+1.

$$\tilde{\mathbf{K}}_{L+1,n} = \begin{bmatrix} 0 \\ \tilde{\mathbf{K}}_{L,n-1} \end{bmatrix} - \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} \quad (1.37)$$

$$\tilde{\mathbf{K}}_{L+1,n} = \begin{bmatrix} \tilde{\mathbf{K}}_{L,n} \\ 0 \end{bmatrix} - \frac{\bar{r}_{L,n}}{\lambda\beta_{L,n-1}} \begin{bmatrix} -\mathbf{b}_{L,n-1} \\ 1 \end{bmatrix} \quad (1.38)$$

$\bar{e}_{L,n}$: Erreur de prédiction d'aller.

$\bar{r}_{L,n}$: Erreur de prédiction de retour.

Si nous éliminons $\tilde{\mathbf{K}}_{L+1,n}$ dans les deux dernières équations nous obtenons la forme récursive suivante pour le gain de Kalman dual :

$$\begin{bmatrix} \tilde{\mathbf{K}}_{L,n} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{K}}_{L,n-1} \end{bmatrix} - \frac{\bar{e}_{L,n}}{\lambda\alpha_{L,n-1}} \begin{bmatrix} 1 \\ -\mathbf{a}_{L,n-1} \end{bmatrix} + \frac{\bar{r}_{L,n}}{\lambda\beta_{L,n-1}} \begin{bmatrix} -\mathbf{b}_{L,n-1} \\ 1 \end{bmatrix} \quad (1.39)$$

La mise à jour récursive des prédicteurs ‘aller’ et ‘retour’ de l’algorithme FRLS est donnée par :

$$\mathbf{a}_{L,n} = \mathbf{R}_{L,n-1}^{-1} \mathbf{P}_{L,n}^f \quad (1.40)$$

$$\mathbf{P}_{L,n}^f = \sum_{i=1}^n \lambda^{n-i} \mathbf{X}_{L,i-1} x_i$$

$$\mathbf{P}_{L,n}^b = \sum_{i=1}^n \lambda^{n-i} \mathbf{X}_{L,i} x_{i-L}$$

D’où :

$\mathbf{P}_{L,n}^f$: Estimation de la corrélation croisée de x_n avec sa valeur $\mathbf{X}_{L,n-1}$.

$\mathbf{P}_{L,n}^b$: Estimation de la corrélation croisée de x_{n-L} avec sa valeur $\mathbf{X}_{L,n}$.

Pour un signal d’entrée stationnaire, les variances des prédicteurs ‘aller’ et ‘retour’ sont égales, avec les mêmes coefficients d’ordre inversé.

L’algorithme FRLS se résume dans le tableau suivant :

Initialisation : $\mathbf{a}_{L,0} = \mathbf{b}_{L,0} = \tilde{\mathbf{K}}_0 = \mathbf{0}_L$; $\gamma_{L,0}=1$; $\alpha_{L,0}=\lambda^L \mathbf{E}_0$; $\beta_{L,0} = \mathbf{E}_{L,0}$; $\mathbf{w}_{L,0}=\mathbf{0}$;

Partie prédiction :

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

$$\bar{r}_{L,n} = \mathbf{x}_{n-L} - \mathbf{b}_{L,n-1}^T \mathbf{X}_{L,n};$$

$$\alpha_{L,n} = \lambda \alpha_{L,n-1} + \gamma_{L,n-1} \bar{\mathbf{e}}_n \mathbf{e}_n;$$

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

$$\mathbf{a}_{L,n} = \mathbf{a}_{L,n-1} - \tilde{\mathbf{K}}_{L,n-1} \gamma_{L,n-1} \bar{\mathbf{e}}_{L,n};$$

$$\begin{bmatrix} \tilde{\mathbf{K}}_{L,n} \\ 0 \end{bmatrix} = \tilde{\mathbf{K}}_{L+1,n} - \tilde{\mathbf{K}}_{L+1,n}^{L+1,n} \begin{bmatrix} -\mathbf{b}_{L,n-1} \\ \mathbf{I} \end{bmatrix}$$

$$\varepsilon_{L,n} = \bar{r}_{L,n} + \lambda^{-L+1} \gamma_{L,n-1} \alpha_{L,n-1} \tilde{\mathbf{K}}_{L+1,n}^{L+1,n}$$

$$r_{L,n}^{-b} = \bar{r}_{L,n} + \mu^b \varepsilon_{L,n}$$

$$\gamma_{L,n} = \frac{\mathbf{I}}{\mathbf{I} - \tilde{\mathbf{K}}_{L,n}^T \mathbf{X}_{L,n}}$$

$$\mathbf{b}_{L,n} = \mathbf{b}_{L,n-1} - r_{L,n}^{-b} \gamma_{L,n} \tilde{\mathbf{K}}_{L,n}$$

Partie filtrage :

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

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \tilde{\mathbf{K}}_{L,n} \bar{\varepsilon}_{L,n} \gamma_{L,n};$$

Tableau 1-4: Résumé de L'algorithm FRLS

1.4.4 les critères de comparaison des algorithmes adaptatifs :

Critères	Définitions
Vitesse de convergence	C'est le Nombre d'itération pour atteindre le régime permanent. La vitesse de la famille des algorithmes des gradients stochastiques dépend de la dynamique spectrale du signal(degré de corrélation) par contre la famille

	des algorithmes des moindres carrées est indépendante de cette dynamique spectrale.
Erreur quadratique moyenne	plus l'erreur de filtrage en régime permanent est faible plus le système s'améliore.
Capacité de poursuite	Le filtrage adaptatif doit suivre les variations de la réponse impulsionnelle (le mouvement des personnes dans la salle)
Complexité de calcul	Représente Le nombre d'opérations arithmétiques par itération.

Tableau 1-5: Les critères de comparaison des algorithmes adaptatifs.

1.5 Conclusion :

Le contexte de l'étude a été présenté en commençant par un détail sur l'origine et l'annulation de l'écho acoustique, le principe de filtrage adaptatif est expliqué ultérieurement par la suite les critères de comparaison des algorithmes adaptatifs ont été décrits. Enfin une partie a été consacré pour définir les différentes familles des algorithmes adaptatifs.

Chapitre 2

2. CHAPITRE 2 : Algorithmes NLMS Optimisé (OP-NLMS)

2.1 Introduction :

L'algorithme adaptatif, noyau du filtrage adaptatif, permet de mettre à jour les coefficients du filtre pour converger vers une solution optimale, les algorithmes de gradient stochastique (LMS et ses variantes) sont les plus employés dans le domaine d'annulation d'écho acoustique. Ces algorithmes sont caractérisés par leurs stabilités numériques, et simplicités d'implémentations, cependant la présence du bruit additif affecte les performances de ces derniers. Afin de remédier à ce problème de nouvelles versions optimisées sont proposées.

Dans ce chapitre, nous avons abordé respectivement les algorithmes NLMS optimisée "OP-NLMS" et le NLMS optimisé récursif "OP-NLMS-REC".

2.2 L'algorithme OP-NLMS :

Les algorithmes LMS et NLMS ont un souci d'optimisation conjointe sur les paramètres normalisés de deux facteurs importants. Le premier concerne le pas d'adaptation μ qui représente un compromis entre le taux de convergence et le mauvais réglage de la puissance de l'erreur du filtrage finale. Le second dépend de la constante de régularisation c_0 conditionnée pour la résolution des problèmes mal posés. La méthode d'optimisation intervient pour réformer le comportement de l'algorithme NLMS en existence de bruit additif sur le signal désiré.

Ces deux paramètres peuvent être assurés, afin d'arranger le conflit entre la convergence rapide et le mauvais ajustement [6].

2.2.1 L'étude de convergence de l'algorithme NLMS [9 , 10]:

La convergence s'effectue à partir d'une analyse déterminée, nous supposons \mathbf{w}_n un vecteur aléatoire de moyenne nulle, qui suit un modèle simplifié du premier ordre donné par l'équation suivante :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{z}_n \quad (2.1)$$

$$\mathbf{z}_n = \mathbf{w}_n - \mathbf{w}_{n-1} \quad (2.2)$$

Sachant que :

\mathbf{z}_n : Vecteur signal du bruit blanc Gaussien de moyenne nulle, non corrélé à \mathbf{w}_{n-1} .

$\mathbf{R}_z = \sigma_z^2 \mathbf{I}_L$: La matrice de corrélation de \mathbf{z}_n .

\mathbf{I}_L : Matrice d'identité $L * L$.

σ_z^2 : Variance des incertitudes dans \mathbf{h}_n .

L'équation (2.1) définit un modèle à variable d'états. établit pour développer l'analyse de convergence de l'algorithme NLMS, ce dernier est considéré avec la mise à jour :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \frac{\mathbf{x}_n \bar{\varepsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + c_0} \quad (2.3)$$

$$\bar{\varepsilon}_n = d_n - \mathbf{x}_n^T \mathbf{w}_{n-1} \quad (2.4)$$

$\bar{\varepsilon}_n$: Erreur du filtre adaptatif.

Le désalignement à posteriori est défini comme suit:

$$\mathbf{c}_n = \mathbf{h}_n - \mathbf{w}_n \quad (2.5)$$

\mathbf{h}_n : Réponse impulsionnelle du filtre adaptatif.

Par conséquent, le développement de l'équation (2.3) permet d'avoir :

$$\mathbf{c}_n = \mathbf{c}_{n-1} + \mathbf{z}_n - \mu \frac{\mathbf{x}_n \bar{\epsilon}_n}{\mathbf{x}_n^T \mathbf{x}_n + c_0} \quad (2.6)$$

Sachant que :

$$E[\mathbf{x}_n^T \mathbf{x}_n] \approx L\sigma_x^2 \quad (2.7)$$

σ_x^2 : Variance de signal pour les grandes valeurs de L ($L \gg 1$).

Pour des grandes valeurs de L, nous utilisons l'approximation :

$$\mathbf{x}_n^T \mathbf{x}_n \approx L\sigma_x^2 \quad (2.8)$$

Donc :

$$\frac{\mu}{\mathbf{x}_n^T \mathbf{x}_n + c_0} \approx \frac{\mu}{L\sigma_x^2 + c_0} \quad (2.9)$$

Ce terme possède à la fois les paramètres de contrôle μ et c_0 , ainsi que les informations statistiques sur le signal d'entrée, néanmoins nous pouvons le qualifier comme une quantité déterministe, et cela pour une valeur élevée de L et un certain degré de stationnarité du signal d'entrée. Dans ces conditions, nous prenons la norme l_2 dans (2.6), ensuite les espérances mathématiques des deux cotés en utilisant la formule (2.9), enfin nous éliminons les produits non corrélés, nous aurons :

$$\begin{aligned} E[\|\mathbf{c}_n\|_2^2] &= E[\|\mathbf{c}_{n-1}\|_2^2] + L\sigma_x^2 \\ &\quad - \frac{2\mu}{c_0 + L\sigma_x^2} E[\mathbf{x}_n^T \mathbf{c}_{n-1} \bar{\epsilon}_n] \\ &\quad - \frac{2\mu}{c_0 + L\sigma_x^2} E[\mathbf{x}_n^T \mathbf{z}_n \bar{\epsilon}_n] \\ &\quad + \frac{\mu^2}{(c_0 + L\sigma_x^2)^2} E[\mathbf{x}_n^T \mathbf{x}_n \bar{\epsilon}_n^2] \end{aligned} \quad (2.10)$$

Si nous nous concentrons sur les trois derniers termes de (2.10) et en se basant sur (1.9) et (2.1), nous déduisons que le signal d'erreur de (2.4) peut se réécrire sous la forme :

$$\bar{\varepsilon}_n = \mathbf{x}_n^T \mathbf{x}_n c_{n-1} + \mathbf{x}_n^T \mathbf{z}_n + v_n \quad (2.11)$$

D'après la simplification de l'expression (2.10) et sachant que :

$$m_n = E[\|c_n\|_2^2] \quad (2.12)$$

Nous obtenons :

$$m_n = f(\mu, c_0, L, \sigma_x^2) m_{n-1} + g(\mu, c_0, L, \sigma_x^2, \sigma_v^2, \sigma_z^2) \quad (2.13)$$

$$f(\mu, c_0, L, \sigma_x^2) = 1 - \frac{2\sigma_x^2}{L\sigma_x^2 + c_0} \mu + \frac{(L+2)\sigma_x^4}{(L\sigma_x^2 + c_0)^2} \mu^2 \quad (2.14)$$

$$g(\mu, c_0, L, \sigma_x^2, \sigma_v^2, \sigma_z^2) = \frac{\mu^2 L \sigma_x^2 [\sigma_v^2 + (L+2)\sigma_x^2, \sigma_z^2]}{(L\sigma_x^2 + c_0)^2} - \frac{2\mu L \sigma_x^2 \sigma_z^2}{(L\sigma_x^2 + c_0)} + L\sigma_z^2 \quad (2.15)$$

Le résultat de l'expression (2.15) illustre une « séparation » entre les composants de convergence et de dérèglement, donc le terme $f(\mu, c_0, L, \sigma_x^2)$ influence le taux de convergence de l'algorithme.

Comme prévu, cela dépend des paramètres suivants :

- La valeur du pas normalisé.
- La constante de régularisation.
- Le longueur du filtre.
- La puissance du signal d'entrée.

Nous notons que cela ne dépend pas de la puissance de bruit du système σ_v^2 et ou σ_z^2 des incertitudes.

2.2.2 Dérivation de l'algorithme OP-NLMS [9, 10]:

L'exigence idéale de tout algorithme adaptatif dépend de la convergence rapide et des faibles désajustements, ce qui exige le contrôle des paramètres μ et c_0 , car ces derniers influencent sur la performance globale de l'algorithme NLMS, d'autre part un suivi doit être effectué pour résoudre le problème de minimisation en terme de désalignement du système.

Ainsi, en suivant (2.13) nous aurons :

$$\frac{\partial m_n}{\partial \mu} = 0, \frac{\partial m_n}{\partial c_0} = 0 \quad (2.16)$$

D'après la simplification des calculs, les deux équations nous ont permis d'aboutir au même résultat.

$$\frac{\mu_n}{L\sigma_x^2 + c_0} = \frac{m_{n-1} + L\sigma_z^2}{l\sigma_v^2 + (L+2)\sigma_x^2[m_{n-1} + L\sigma_z^2]} \quad (2.17)$$

Ces analyses, ont proposé un processus d'optimisation conjoint avec une estimation correcte de ses paramètres. Le terme figurant dans la partie droite de (2.17) agit comme une taille de pas variable, cela permet d'introduire (2.17) dans (2.3) .

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{[m_{n-1} + L\sigma_z^2]\mathbf{x}_n\bar{\varepsilon}_n}{l\sigma_v^2 + (L+2)\sigma_x^2[m_{n-1} + L\sigma_z^2]} \quad (2.18)$$

Il est nécessaire de mettre à jour le paramètre m_n dans (2.18) . En remplaçant (2.17) dans (2.16), suivi de plusieurs calculs simples , il en résulte :

$$m_n = \left\{ 1 - \frac{\sigma_x^2[m_{n-1} + L\sigma_z^2]}{l\sigma_v^2 + (L+2)\sigma_x^2[m_{n-1} + L\sigma_z^2]} \right\} [m_{n-1} + L\sigma_z^2] \quad (2.19)$$

L'algorithme OP-NLMS est donné ci-dessous dans le tableau 2.1 :

Initialisation :

L=la taille de filtre

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

$$\mathbf{m}_0 = \varepsilon > \mathbf{0}$$

$$\sigma_{z,0}^2 = \mathbf{0}$$

Paramètre :

σ_v^2 : puissance de bruit connue ou estimée.

Pour l'indice de temps $n=1,2,\dots$

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

$$\mathbf{p}_n = \mathbf{m}_{n-1} + L\sigma_{z,n-1}^2$$

$$\sigma_{x,n}^2 = \frac{1}{L} \mathbf{x}_n^T \mathbf{x}_n$$

$$q_n = \frac{p_n}{L\sigma_v^2 + (L+2)p_n\sigma_{x,n}^2}$$

$$\mathbf{u}_n = q_n \mathbf{x}_n \bar{\mathbf{e}}_n$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{u}_n$$

$$\mathbf{m}_n = [1 - q_n\sigma_{x,n}^2] \mathbf{p}_n$$

$$\sigma_{b,n}^2 = \frac{1}{L} \mathbf{u}_n^T \mathbf{u}_n$$

Tableau 2-1: Résumé de l'algorithme OP-NLMS.

2.3 L'algorithme OPNLMS récursif [6]:

L'algorithme NLMS optimisé récursif est proposé afin de simplifier les calculs de l'algorithme NLMS optimisé, selon les modifications apportées sur σ_x^2 et σ_w^2 .

La formule récursive ci-dessous sert à estimer l'énergie :

$$\sigma_{x,n}^2 = \lambda \sigma_{x,n-1}^2 + (1 - \lambda) \mathbf{x}_n^2 \quad (2.20)$$

λ : facteur d'oubli exponentiel.

$$\lambda = 1 - \frac{1}{L} \quad (2.21)$$

La simplification du calcul de la variance σ_z^2 est établie par l'expression suivante :

$$\sigma_{z,n}^2 = \frac{1}{L} \mathbf{u}_n^T \mathbf{u}_n = (q_n \bar{\varepsilon}_n)^2 \mathbf{x}_n^T \mathbf{x}_n \quad (2.22)$$

$$\sigma_{z,n}^2 = \frac{1}{L} (q_n \bar{\varepsilon}_n)^2 \sigma_{x,n}^2 \quad (2.23)$$

Un paramètre à pas fixe μ_2 est ajouté pour que l'algorithme OP-NLMS atteigne une meilleure performance .

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu_2 \mathbf{u}_n \quad (2.24)$$

L'algorithme OP-NLMS-REC est donné ci-dessous dans la table 2-2 :

Initialisation :

L : la taille e filtre

Pour l'indice de temps $n = 1, 2, \dots$

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

$$\mathbf{m}_0 = \varepsilon > \mathbf{0}$$

$$\sigma_{z,0}^2 = \mathbf{0}$$

Paramètres :

σ_v^2 : puissance de bruit connue ou estimée.

Pour l'indice de temps $n=1, 2, \dots$

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

$$\mathbf{p}_n = \mathbf{m}_{n-1} + L \sigma_{z,n-1}^2$$

$$\sigma_{x,n}^2 = \lambda \sigma_{x,n-1}^2 + (1 - \lambda) \mathbf{x}_n^T \mathbf{x}_n$$

$$q_n = \frac{\mathbf{p}_n}{L \sigma_v^2 + (L + 2) \mathbf{p}_n \sigma_{x,n}^2}$$

$$\mathbf{u}_n = q_n \mathbf{x}_n \bar{\varepsilon}_n$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu_2 \mathbf{u}_n$$

$$\mathbf{m}_n = [1 - q_n \sigma_{x,n}^2] \mathbf{p}_n$$

$$\sigma_{z,n}^2 = (q_n \bar{\varepsilon}_n)^2 \sigma_{x,n}^2$$

Tableau 2-2: Résumé de l'algorithme OP-NLMS-REC.

2.4 Complexité des calculs :

Le tableau ci-joint évalue la complexité en nombre de multiplications des différents algorithmes :

Les algorithmes	Le nombre de multiplications
NLMS	$2N+3$
NLMS optimisé	$4N+7$
NLMS optimisé récursif	$3N+12$ ou $2N+12$ pour $\mu_2=1$

Tableau 2-3: complexité de calculs.

2.5 Conclusion :

Dans ce chapitre, nous avons présenté dans un premier temps, l'algorithme OP-NLMS , en commençant par l'étude de sa convergence et en finissant par sa dérivation . L'algorithme OP-NLMS-REC est traité ultérieurement, une attention particulière à été accordé à la complexité des calculs. Dans le prochain chapitre, nous proposerons une technique de blanchiment qui améliore les performances de l'algorithme OP-NLMS.

Chapitre 3

3. CHAPITRE 3 :Algorithme OPNLMS avec blanchiment

3.1 Introduction :

L'algorithme de gradient stochastique normalisé (NLMS) a connu un véritable essor, grâce à son arrivée à une maturité aussi bien concernant le développement et en raison de sa faible complexité de calcul et son comportement robuste, l'inconvénient est que ce type d'algorithme souffre d'une convergence lente lorsque la dynamique spectrale est importante, afin de réduire cette dernière nous avons pensé à appliquer une technique blanchissante pour le signal [11].

3.2 Principe du blanchiment :

Le pré-blanchiment est une opération qui traite une séquence de données afin de la rendre statistique comme le bruit blanc. Le « pré » signifie que le blanchiment est précédé par une autre analyse qui fonctionne correctement seulement si le bruit additif est blanc. Ces analyses peuvent être visualisées dans le domaine temporel ou bien fréquentiel, pour cela on doit :

- Faire en sorte que l'auto corrélation du signal d'entrée s'approche le plus possible d'une fonction Dirac.
- Rendre le spectre du signal d'entrée le plus plat que possible.

Toutes les méthodes de blanchiment de la littérature qui ont été proposées dans le cadre de l'algorithme NLMS sont basées sur le principe de réduction de la dynamique spectrale du signal d'entrée $x(n)$ par une prédiction linéaire. Au cours de ce chapitre nous allons décrire quelques méthodes pour améliorer la vitesse de convergence de l'algorithme NLMS. La transposition de ces méthodes dans le cas de l'algorithme OPNLMS n'est pas évidente. A la fin, nous allons étendre la méthode de l'algorithme Fast NLMS [11] à l'algorithme OPNLMS pour améliorer ses performances.

3.3 Résumé de méthodes existantes:

En 1991 [12], les auteurs ont proposé trois structures pour accélérer la vitesse de convergence de l'algorithme LMS. La première structure illustrée dans (Figure. 3.1) insère un filtre de blanchiment P nécessitant un filtre inverse :

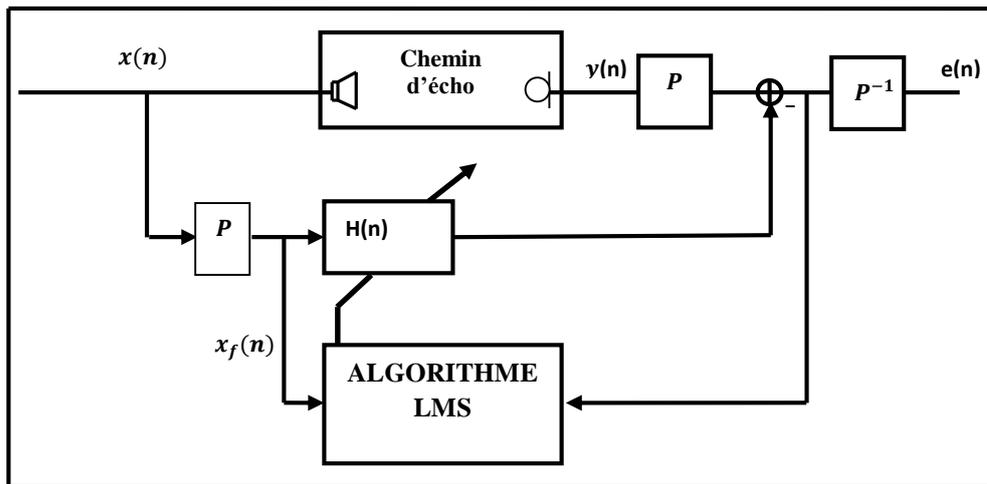


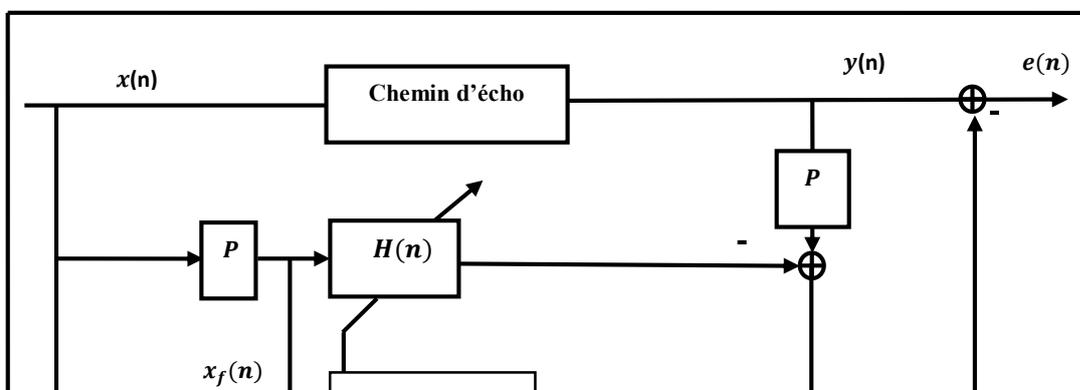
Figure 3-1: Pré-blanchiment sur le chemin d'écho.

La sortie du filtre blanchissant P représente l'erreur de prédiction de l'entrée x qui est notée $x_f(n)$:

$$x_f(n) = x(n) - \sum_{i=1}^P p_i x(n-i) \quad (3.1)$$

p_i : coefficients de prédiction.

La deuxième structure présentée sur la (figure 3.2) élimine le filtre inverse, mais nécessite la copie de l'anneau pour pouvoir en déduire l'erreur $e(n)$ qui est l'écho atténué.



Cette structure est aussi proposée pour l'annulation d'écho acoustique par [13].

La troisième structure proposée par la figure (3.3) est la plus intéressante pour les applications d'écho acoustique, elle évite l'utilisation du prédicteur sur le chemin d'écho, en employant un prédicteur adaptatif $P(n)$ avec l'algorithme d'adaptation de $H(n)$ seulement. Comme la parole est un signal non stationnaire donc $P(n)$ s'adapte aux statistiques à court-terme du signal.

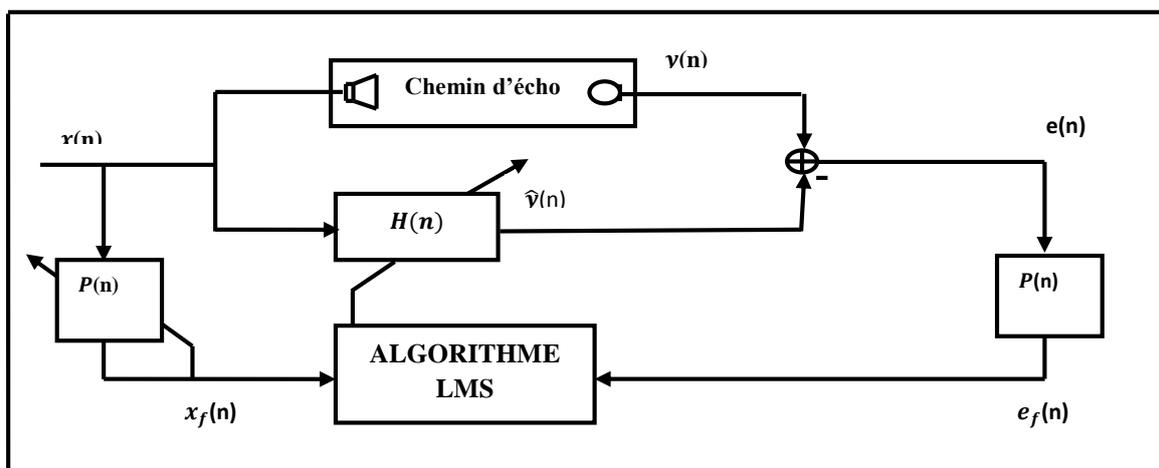


Figure 3-3: Blanchiment des signaux de l'algorithme.

Contrairement aux structures connues une amélioration importante des propriétés transitoires de l'algorithme LMS a été obtenue, tout en assurant un comportement asymptotique identique.

Des structures similaires à celle de la figure (3.3) ont été proposées dans [14] :

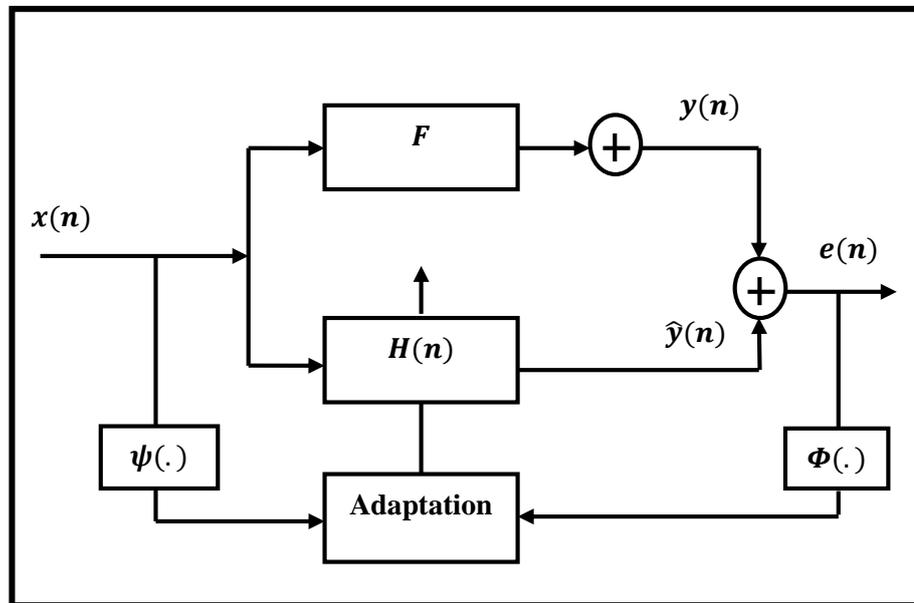


Figure 3-4: structure à base d'algorithmes de signe.

L'équation d'adaptation du filtre est donnée par :

$$H(n + 1) = H(n) + \mu\Phi(e(n))\psi(X(n)) \quad (3.2)$$

La famille des algorithmes de signe constitue une variante de la famille de gradient stochastique, cette approche a été proposée afin de réduire la complexité des calculs. L'introduction de la structure blanchissante sur cette famille suscite une étude comparative de plusieurs variantes d'algorithmes de signe:

1. sans blanchiment de l'entrée et sans filtrage de l'erreur (classique).
2. sans blanchiment de l'entrée mais avec filtrage de l'erreur.
3. avec blanchiment de l'entrée et sans filtrage de l'erreur.
4. avec blanchiment de l'entrée et filtrage de l'erreur.

La troisième structure est la meilleure en terme de vitesse de convergence [14].

Le filtrage adaptatif en sous bandes [15,16] est une technique intéressante pour réduire la dynamique spectrale du signal $x(n)$ qui permet une amélioration au niveau de vitesse de convergence du NLMS.

Le rôle de l'annuleur d'écho adaptatif en sous-bande est d'éliminer l'écho dans différentes bandes de fréquence en les séparant par un ensemble de filtres adaptatifs parallèles (figure 3.5), chacun de ces filtres identifie seulement le chemin d'écho dans une petite partie de la largeur de bande originale. Chaque sous bande couvre seulement une petite fraction du spectre du signal d'entrée, ce qui permet aux filtres adaptatifs en sous-bande d'opérer sur les signaux avec des spectres plus plats par rapport au signal original de la bande sans sous bandes, ayant un taux de convergence sensiblement plus rapide.

Par ailleurs, les filtres adaptatifs en sous-bande fonctionnent à un taux d'échantillonnage plus faible, rapportant une réduction significative de la complexité des calculs.

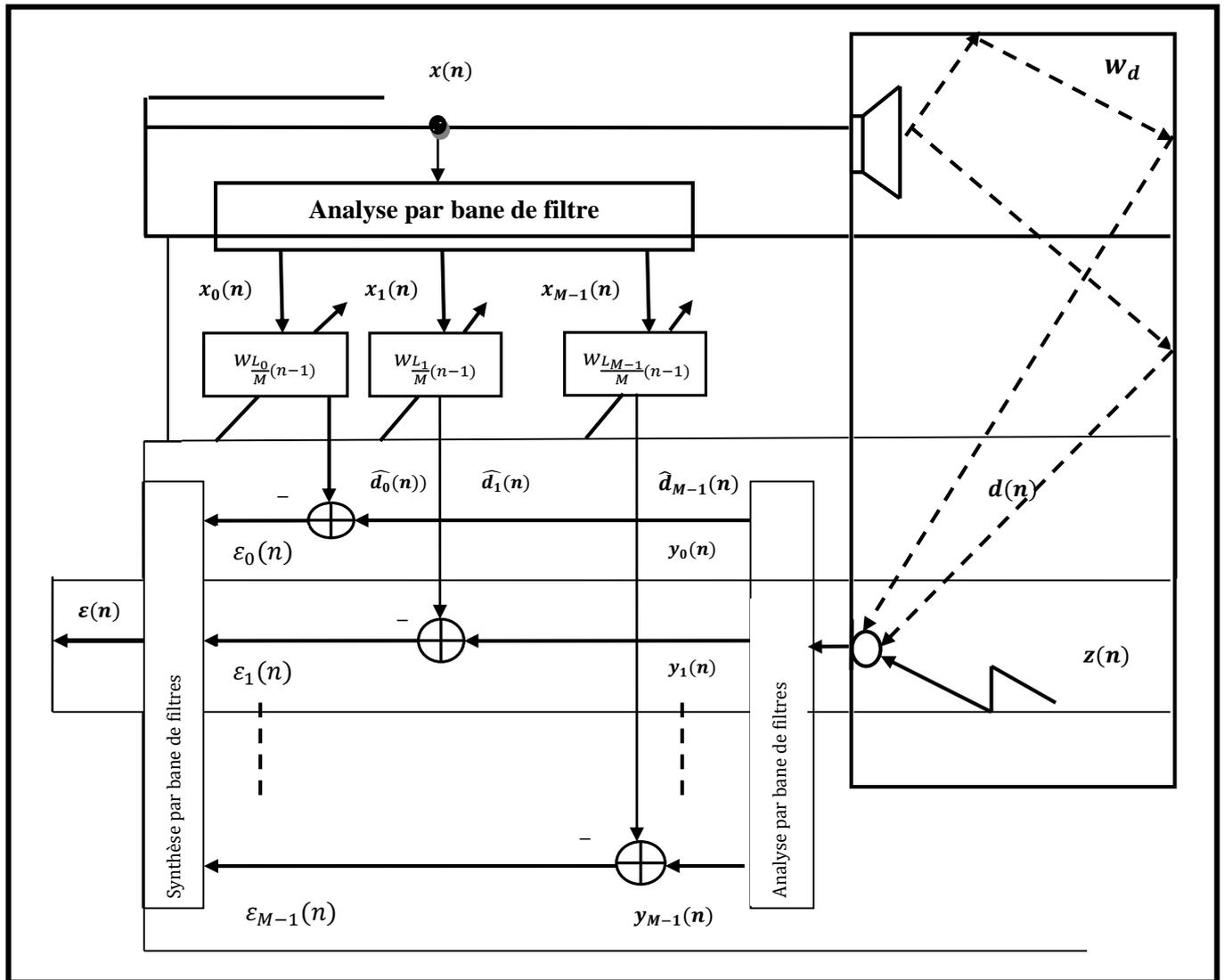


Figure 3-5: Filtrage adaptatif en sous bandes.

En [2004], les auteurs ont proposé de combiner le filtre FIR avec une prédiction linéaire dans une structure en échelle (Treillis) comme est montré dans la (figure 3.6) dont les signaux sont décrits comme suit :

n : indice temporel discret.

$s(n)$: signal $x(n)$.

$y(n)$: l'écho estimé.

$f(j, n)$: erreurs de prédiction aller (forward).

$b(j, n)$: erreurs de prédiction retour (backward).

$K(j, n)$: coefficients de réflexion, appliqués aussi coefficients PARCOR.

$G(j, n)$: Filtre transverse en relation avec le chemin d'écho.

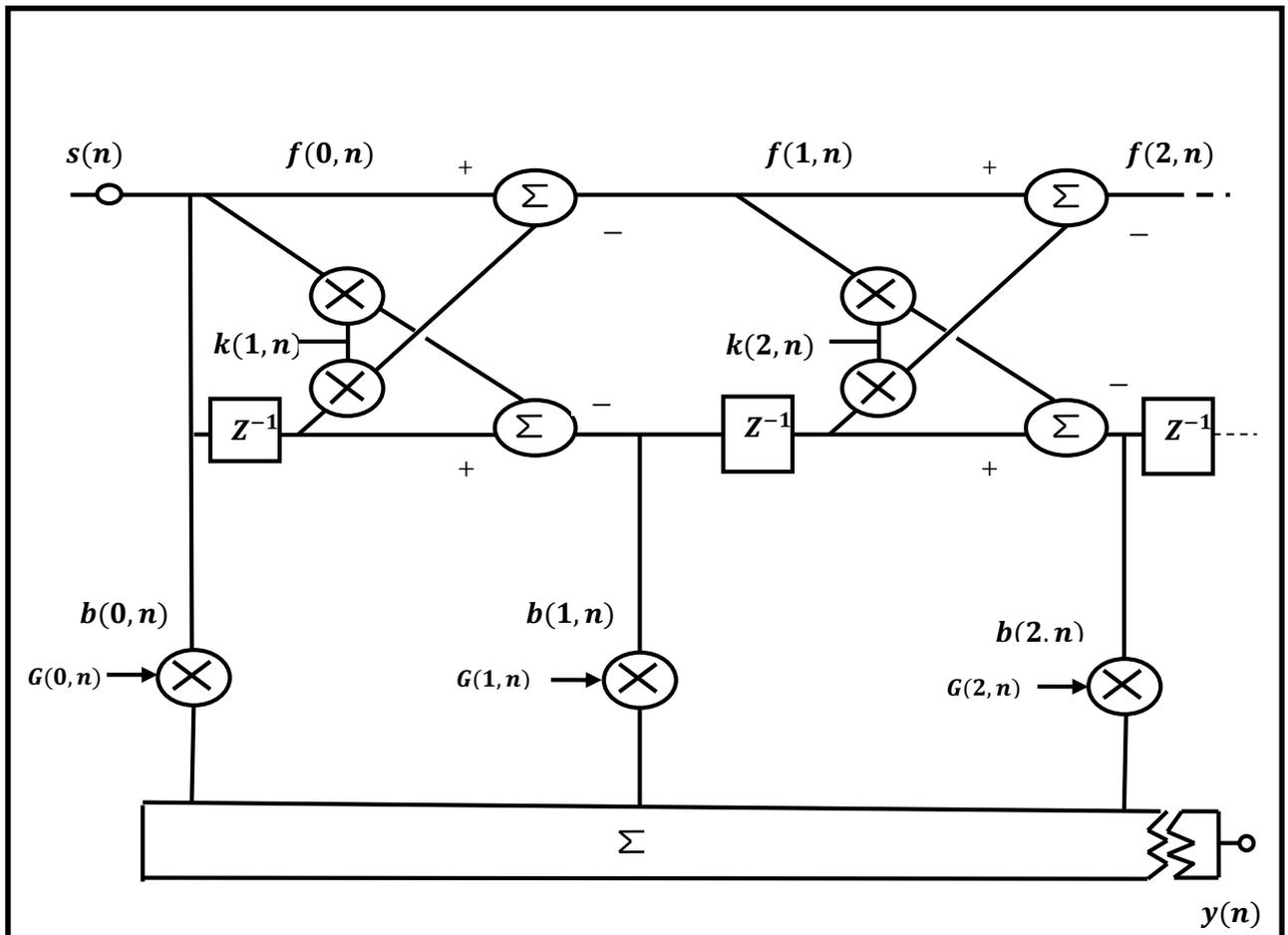


Figure 3-6: Filtrage adaptatif en treillis.

L'intérêt de cette structure est que la partie treillis réalise un blanchiment du signal d'entrée en parallèle avec un filtre FIR de coefficients $G(j, n)$. Le principal inconvénient de cette structure est que les coefficients du filtre FIR $G(j, n)$ ne sont pas en relation directe avec les coefficients du FIR du canal acoustique H . Pour cette raison, les auteurs ont proposé une structure plus simplifiée avec une cellule en treillis (prédiction d'ordre 1) suivi par un filtre FIR avec accès directe aux coefficients du filtre du canal acoustique H (figure 3.7).

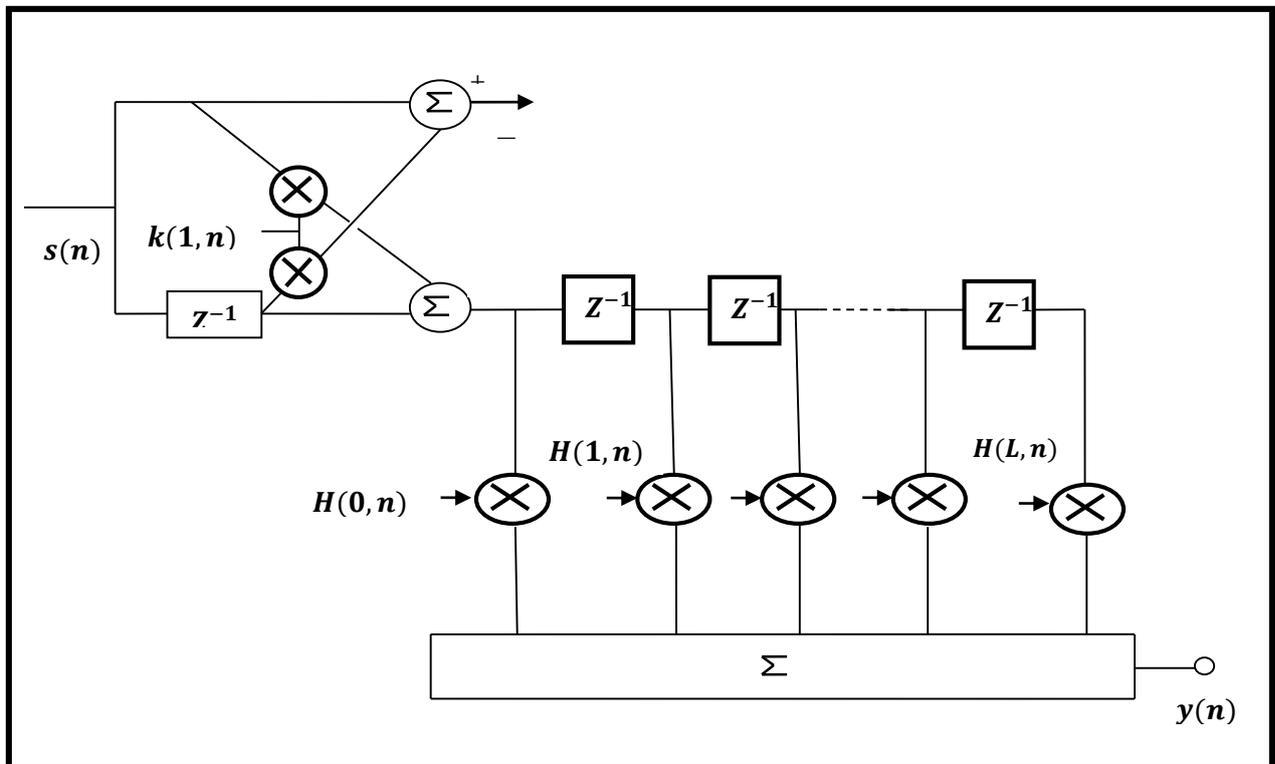


Figure 3-7: Structure treillis simplifiée.

En 2013 [11] : les auteurs ont présenté un nouvel algorithme de type NLMS à convergence rapide obtenu à partir d'un algorithme moindre carré rapide MCR où le gain (Kalman) d'adaptation est calculé en annulant complètement des prédicteurs aller et retour de l'algorithme MCR :

$$\begin{bmatrix} \tilde{\mathbf{k}}_L(n) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_L(n-1) \end{bmatrix} - \frac{\bar{e}_L(n)}{\lambda \alpha_L(n-1)} \begin{bmatrix} 1 \\ -\mathbf{a}_L(n-1) \end{bmatrix} + \frac{\bar{r}_L}{\lambda \beta_L(n-1)} \begin{bmatrix} -\mathbf{b}_L(n-1) \\ 1 \end{bmatrix} \quad (3.3)$$

$\bar{e}_L(n)$: Erreur de prédiction aller

$\alpha_L(n)$: Variance d'erreur de prédiction aller .

\bar{r}_L : Erreur de prédiction retour.

$\beta_L(n)$: Variance d'erreur de prédiction retour.

λ : Facteur d'oubli exponentiel.

Nous montrons que lorsque $\mathbf{a}_L(n) = \mathbf{b}_L(n) = 0$, le dual Kalman gain (3.3) s'écrit :

$$\begin{bmatrix} \tilde{\mathbf{k}}_L(n) \\ \mathbf{k}(n) \end{bmatrix} = \begin{bmatrix} -\frac{x(n)}{\lambda\alpha_L(n-1)} \\ \tilde{\mathbf{k}}_L(n-1) \end{bmatrix} \quad (3.4)$$

Où la variable $k(n)$ est la quantité $-\frac{\bar{r}_L}{\lambda\beta_L(n-1)}$ et $\bar{e}_L(n) = x(n)$.

Dans l'algorithme proposé, nous avons éliminé complètement les prédicteurs pour appliquer seulement une erreur de prédiction "aller" notée $e(n)$ pour évaluer le gain de Kalman dual :

$$\begin{bmatrix} \tilde{\mathbf{k}}_L(n) \\ \mathbf{k}(n) \end{bmatrix} = \begin{bmatrix} -\frac{e(n)}{\lambda\alpha(n-1) + c_0} \\ \tilde{\mathbf{k}}_L(n-1) \end{bmatrix} \quad (3.5)$$

Où $\tilde{\mathbf{k}}_L(n)$ est le gain de Kalman dual.

L'erreur de prédiction $e(n)$ introduit un blanchiment au niveau du signal $x(n)$. la génération du gain d'adaptation peut être obtenue par une prédiction linéaire d'ordre P.

Par souci de simplicité les auteurs ont proposé d'utiliser un prédicteur récursif d'ordre 1 ($P =$

1). L'adaptation du filtre FIR se résume dans les deux équations suivantes :

$$\varepsilon_N = d(n) - \mathbf{h}_L^T(n-1)X_L(n) \quad (3.6)$$

$$\mathbf{h}_L(n) = \mathbf{h}_L(n-1) - \mu \varepsilon_L(n) \gamma_L(n) \tilde{\mathbf{k}}_L(n) \quad (3.7)$$

Dans les paragraphes qui suivent, l'idée de cet algorithme fast NLMS sera étendue à l'algorithme OPNLMS afin d'obtenir un algorithme NLMS optimisé en terme de vitesse de convergence et de l'erreur finale.

3.4 La prédiction linéaire :

Un modèle de prédiction linéaire prévoit l'amplitude d'un signal à l'instant n , $x(n)$, en utilisant une combinaison pondérée de P anciens échantillons $[x(n-1), x(n-2), \dots, x(n-P)]$ comme :

$$\hat{x}(n) = \sum_{k=1}^P a_k x(n-k) \quad (3.8)$$

n : indice de temps discret .

$\hat{x}(n)$: prédiction de $x(n)$.

a_k : coefficients du prédicteur .

Une implémentation schématique du prédicteur de l'équation (3.8) est illustrée sur la figure (3.8). L'erreur de prédiction $e(n)$ est définie comme la différence entre la valeur réelle de l'échantillon $x(n)$ et sa valeur prédite $\hat{x}(n)$.

$$\begin{aligned} e(n) &= x(n) - \hat{x}(n) \\ &= x(n) - \sum_{k=1}^P a_k x(n-k) \end{aligned} \quad (3.9)$$

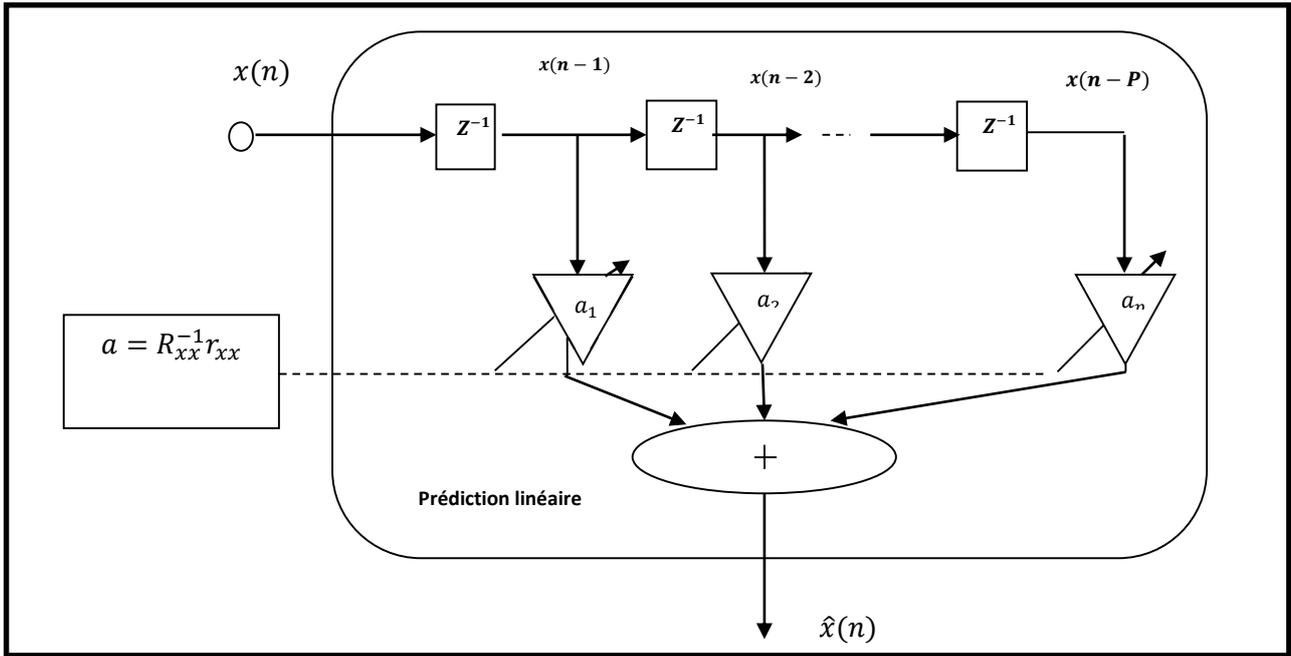


Figure 3-8: Prédiction linéaire.

Pour un signal parole, un ordre P de 10 à 12 coefficients est suffisant pour prédire correctement un signal de ce type qui est stationnaire sur des périodes de 20 à 30ms.

Les meilleurs coefficients du prédicteur sont obtenus en minimisant l'erreur quadratique moyenne définie comme suit :

$$E[e^2(n)] = E \left[\left(x(n) - \sum_{k=1}^P a_k x(n-k) \right)^2 \right] \quad (3.10)$$

$$= E[x^2(n)] - 2 \sum_{k=1}^P a_k E[x(n)x(n-k)] + \sum_{k=1}^P a_k \sum_{j=1}^P a_j E[x(n-k)x(n-j)]$$

$$r_{xx}(0) - 2r_{xx}^T a + a^T R_{xx} a$$

Avec :

$R_{xx} = E[x(n)x^T(n)]$: la matrice d'autocorrelation du vecteur d'entrée $x^T(n) = [x(n-1), x(n-2), \dots, x(n-P)]$.

$r_{xx} = E[x(n)\mathbf{x}]$: vecteur d'autocorrelation .

$\mathbf{a}^T = [a_1, a_2, \dots, a_p]$ est le vecteur des coefficients de prédicteur .

D'après l'équation (3.10) , le gradient de l'erreur quadratique moyenne de prédiction par rapport au vecteur de coefficient prédicteur \mathbf{a} est donné par :

$$\frac{\partial}{\partial \mathbf{a}} E[e^2(n)] = -2r_{xx}^T + 2\mathbf{a}^T R_{xx} \quad (3.11)$$

La solution optimale d'une prédiction d'ordre p est donnée sous la forme suivante :

$$\mathbf{a} = \mathbf{R}_{xx}^{-1} r_{xx} \quad (3.12)$$

L'équation (3.12) peut être notée sous forme d'une matrice d'ordre P .

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \dots & r_{xx}(p-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \dots & r_{xx}(p-2) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \dots & r_{xx}(p-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(p-1) & r_{xx}(p-2) & r_{xx}(p-3) & \dots & r_{xx}(0) \end{bmatrix}^{-1} \begin{bmatrix} r_{xx}(0) \\ r_{xx}(1) \\ r_{xx}(2) \\ \vdots \\ r_{xx}(p) \end{bmatrix} \quad (3.13)$$

Avec :

$$r_{xx}(i) = E(x(n)x(n-i)) = r_i \quad (3.14)$$

A partir du prédicteur optimal (3.13), nous déduisons deux prédicteurs d'ordre 1 et 2 que nous allons utiliser par la suite avec l'algorithme OPNLMS.

- **Prédicteur d'ordre 1 :**

A partir de l'équation (3.14), il est facile d'avoir le prédicteur d'ordre 1 :

$$a_1 = \frac{r_{xx}(1)}{r_{xx}(0)} = \frac{r_1}{r_0} \quad (3.15)$$

Par la suite, nous donnerons un estimateur récursif pour ce prédicteur.

- **Le Prédicteur d'ordre 2 :**

Représente l'inverse de la matrice R_{xx} d'ordre 2.

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} r_{xx}(0) & r_{xx}(1) \\ r_{xx}(1) & r_{xx}(0) \end{bmatrix}^{-1} \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \end{bmatrix} \quad (3.16)$$

L'équation des cofacteurs ci-dessus permet de calculer l'inverse des matrices de dimensions 2×2 :

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (3.17)$$

$$A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \quad (3.18)$$

Donc :

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \frac{1}{r_{xx}^2(0) - r_{xx}^2(1)} \begin{bmatrix} r_{xx}(0) & -r_{xx}(1) \\ -r_{xx}(1) & r_{xx}(0) \end{bmatrix} \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \end{bmatrix} \quad (3.19)$$

d : Déterminant de la matrice (non nul).

$$d = r_{xx}^2(0) - r_{xx}^2(1) \quad (3.20)$$

Nous écrivons les solutions sous les formes suivantes :

$$a_1 = \frac{r_0 r_1 - r_1 r_2}{d} = \frac{\frac{r_1}{r_0} - \frac{r_1}{r_0} \times \frac{r_2}{r_0}}{1 - \left(\frac{r_1}{r_0}\right)^2} \quad (3.21)$$

$$a_2 = \frac{r_0 r_2 - r_1^2}{d} = \frac{\frac{r_2}{r_0} - \left(\frac{r_1}{r_0}\right)^2}{1 - \left(\frac{r_1}{r_0}\right)^2} \quad (3.22)$$

Par la suite, nous allons développer des estimateurs récursifs pour les équations du prédicteur d'ordre 2.

3.5 Algorithme OPNLMS-R avec blanchiment :

Dans ce paragraphe, un algorithme OPNLMS-R avec blanchiment est délivré en intégrant un gain d'adaptation similaire au gain de l'algorithme fast NLMS dans l'adaptation du filtre à la place du vecteur signal $x(n)$ [11]. Nous rappelons ici les équations essentielles de l'algorithme OPNLMS-R (chap.2) :

Pour l'indice de temps $n=1,2,\dots$

Erreur de filtrage :

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

$$p_n = m_{n-1} + L\sigma_{z,n-1}^2$$

Energie du signal :

$$\sigma_{x,n}^2 = \lambda\sigma_{x,n-1}^2 + (1 - \lambda)x_n^2$$

$$q_n = \frac{p_n}{L\sigma_v^2 + (L + 2)p_n\sigma_{x,n}^2}$$

Vecteur d'adaptation :

$$\mathbf{u}_n = q_n \mathbf{x}_n \bar{\varepsilon}_n$$

Filtre adaptatif :

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{u}_n \text{ avec } \mu_2 = 1$$

$$m_n = [1 - q_n \sigma_{x,n}^2] p_n$$

$$\sigma_{z,n}^2 = (q_n \bar{\varepsilon}_n)^2 \sigma_{x,n}^2$$

Nous blanchissons le vecteur $x(n)$ du vecteur u_n qui sert à adapter le filtre w_n . Pour cela, un vecteur gain blanchissant remplace $x(n)$:

$$\mathbf{G}_L(n) = [e(n), \mathbf{G}_{L-1}(n-1)] \quad (3.23)$$

Où G_L et G_{L-1} sont respectivement des vecteurs d'ordres L et $L-1$ et $e(n)$ est l'erreur de prédiction donnée par :

$$e(n) = x(n) - \mathbf{A}_P^T(n) \mathbf{x}_n(2:P) \quad (3.24)$$

$\mathbf{A}_P^T(n) = [a_1(n), a_2(n), \dots, a_P(n)]^T$: vecteur de prédiction d'ordre P .

Dans le but d'évaluer ce vecteur trois façons sont proposées par la suite .

Le vecteur \mathbf{u}_n utilisant le vecteur de blanchiment s'écrit alors :

$$\mathbf{u}_n = q_n \mathbf{G}_L(n) \bar{\varepsilon}_n \quad (3.25)$$

L'algorithme OPNLMS-R avec blanchiment est résumé sur le tableau (3-1):

Initialisation :

$$\mathbf{w}_0 = \mathbf{0}_{L \times 1}, \mathbf{G}_L(\mathbf{0}) = \mathbf{0}_{L \times 1}, \mathbf{A}_P(\mathbf{0}) = \mathbf{0}_{P \times 1}$$

$$\mathbf{m}_0 = \varepsilon > \mathbf{0}$$

$$\sigma_{z,0}^2 = \mathbf{0}$$

σ_v^2 : puissance de bruit connue ou estimée.

Prédiction et filtrage :

Pour l'indice de temps $n=1,2,\dots$

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

$$\mathbf{p}_n = \mathbf{m}_{n-1} + L\sigma_{z,n-1}^2$$

$$\sigma_{x,n}^2 = \lambda\sigma_{x,n-1}^2 + (1 - \lambda)\mathbf{x}_n^2$$

$$\mathbf{q}_n = \frac{\mathbf{p}_n}{L\sigma_v^2 + (L + 2)\mathbf{p}_n\sigma_{x,n}^2}$$

Calculer le prédicteur $\mathbf{A}_p^T(n)$ (voir plus loin).

$$\mathbf{e}(n) = x(n) - \mathbf{A}_p^T(n)\mathbf{x}_n(2:P) \text{ (erreur de prédiction).}$$

$$\mathbf{G}_L(n) = [\mathbf{e}(n), \mathbf{G}_{L-1}(n-1)]$$

$$\mathbf{u}_n = \mathbf{q}_n \mathbf{G}_L(n) \bar{\varepsilon}_n$$

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{u}_n$$

$$\mathbf{m}_n = [1 - \mathbf{q}_n\sigma_{x,n}^2]\mathbf{p}_n$$

$$\sigma_{z,n}^2 = (\mathbf{q}_n \bar{\varepsilon}_n)^2 \sigma_{x,n}^2$$

Tableau 3-1: L'algorithme OPNLMS-R avec blanchiment.

Les trois façons qui calculent le prédicteur $\mathbf{A}_p^T(n)$ sont:

- une routine de Matlab.
- un prédicteur récursif d'ordre 1 comme dans l'algorithme fast NLMS .
- un prédicteur récursif d'ordre 2 déduit à partir du prédicteur optimal du paragraphe (3.4).

a) Prédicteur d'ordre P de Matlab.

Le logiciel Matlab offre une routine qui calcule le prédicteur $A_p^T(n)$ dont la syntaxe est donnée par :

$$[\mathbf{pr}, \mathbf{vr}] = \text{lpc}(x_n, \mathbf{P}) \quad (3.26)$$

Avec :

\mathbf{P} : ordre de prédiction.

x_n : vecteur signal d'entrée.

\mathbf{vr} : variance de l'erreur de prédiction.

\mathbf{pr} : vecteur contenant le vecteur de prédiction stocké avec un signe inversé. Le prédicteur réel se déduit alors par $A_p^T(n) = -\mathbf{pr}(2: \text{end})$;

L'inconvénient ici est que nous ignorons la méthode employée dans la routine et sa complexité de calcul. Cette dernière est choisie pour tester l'influence de la taille P du prédicteur.

b) Prédicteur Récursif d'ordre 1 :

Le prédicteur optimal donné par (3.15) doit être estimé de manière récursive comme dans le cas de l'algorithme fast NLMS :

$$a_1(n) = \frac{r_1(n)}{r_0(n) + c_a} \quad (3.27)$$

Les estimateurs récursifs des coefficients de corrélation :

$$r_1(n) = \lambda_a r_1(n-1) + x(n)x(n-1) \quad (3.28)$$

$$r_0(n) = \lambda_a r_0(n-1) + x^2(n) \quad (3.29)$$

$r_0(n)$: Désigne la puissance du signal d'entrée.(de valeur initiale E_0).

$r_1(n)$: premier coefficient de la fonction d'autocorrélation de $x(n)$.(de valeur initiale nulle).

λ_a : facteur d'oubli et c_a une constante de régularisation qui évite les divisions par des zéros.

L'erreur de prédiction $e(n)$ par un prédicteur d'ordre 1 est donnée par :

$$e(n) = x(n) - a_1(n)x(n-1) \quad (3.30)$$

c) Prédicteur Récursif d'ordre 2 :

Le prédicteur optimal d'ordre 2 est donné par les équations (3.13) à (3.16). Les estimateurs récursifs sont exprimés par les relations suivantes :

$$r_i(n) = \lambda_a r_i(n-1) + x(n)x(n-i) \quad i = 0,1,2 \quad (3.31)$$

$$rr_i(n) = \frac{r_i(n)}{r_0(n) + c_a} \quad i = 1,2 \quad (3.32)$$

$$a_1(n) = \frac{rr_1(n) - rr_1(n)rr_2(n)}{1 - rr_1(n)^2} \quad (3.33)$$

$$a_2(n) = \frac{rr_2(n) - rr_1(n)^2}{1 - rr_1(n)^2} \quad (3.34)$$

$r_0(n)$: puissance du signal d'entrée.(de valeur initiale E_0).

$r_1(n)$: premier coefficient de la fonction d'autocorrélation de $x(n)$ avec $r_1(0) = 0$.

$r_2(n)$: deuxième coefficient de la fonction d'autocorrélation de $x(n)$.(de valeur initiale est nulle).

L'erreur de prédiction $e(n)$ pour un prédicteur d'ordre 2 est donnée ci-dessous :

$$e(n) = x(n) - a_1(n)x(n-1) - a_2(n)x(n-2) \quad (3.35)$$

3.6 Conclusion :

Dans ce chapitre, nous avons défini le principe de blanchiment ,puis nous avons présenté quelques méthodes de blanchiment proposées pour l’algorithme NLMS. La transposition de ces techniques dans le cas de l’algorithme OPNLMS n’est pas évidente, pour cette raison nous avons adopté puis proposé un algorithme OPNLMS-BL basé sur la technique de blanchiment qui est développée avec l’algorithme fast NLMS.

Dans le prochain chapitre, nous allons soumettre l’algorithme proposé aux simulations Matlab avec différents signaux et différentes tailles du filtre FIR..

Chapitre 4

4. CHAPITRE 4 : résultats de simulation

4.1 Introduction :

Dans ce chapitre nous avons mis en œuvre un algorithme de filtrage adaptatif nommé ‘‘OP-NLMS-BL’’ proposé dans le chapitre précédent, l’objectif de cette mise en œuvre est de mettre l’accent sur ce nouvel algorithme et de le comparer avec l’algorithme ‘‘OP-NLMS’’ qui a été déjà étudié afin d’atteindre un meilleur rendement au niveau de vitesse de convergence et puissance d’erreur en régime permanent.

4.2 Critère de performance MSE :

La comparaison des algorithmes fait appel au critère de l’évolution temporelle de L’erreur Quadratique Moyenne (MSE) qui est donnée par [6] :

$$EQM(n) = 10\log(\sigma_e^2(n)) \quad (4.1)$$

La puissance moyenne $\sigma_e^2(n)$ est évaluée sur des moyennes temporelles de 256 ou 512 échantillons consécutifs.

4.3 Description des signaux et des paramètres :

Les Signaux :

Les signaux d’entrées utilisés dans nos simulations sont représentés sur les figures (4.1) à (4.3) et sont décrits ci-dessous. La fréquence d’échantillonnage est de 16 kHz et l’amplitude est codée sur 16 bits.

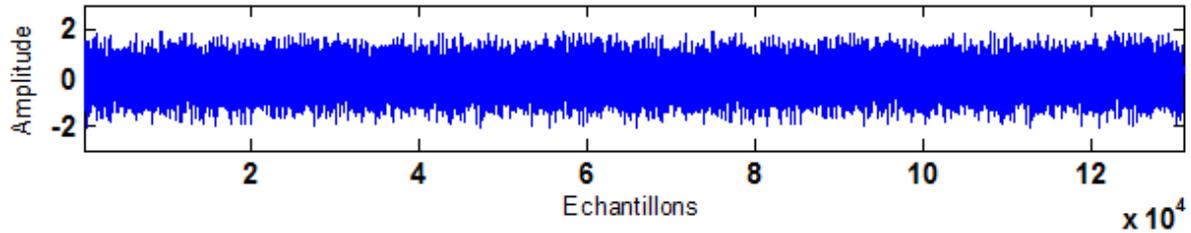


Figure 4-1 : signal usasi.

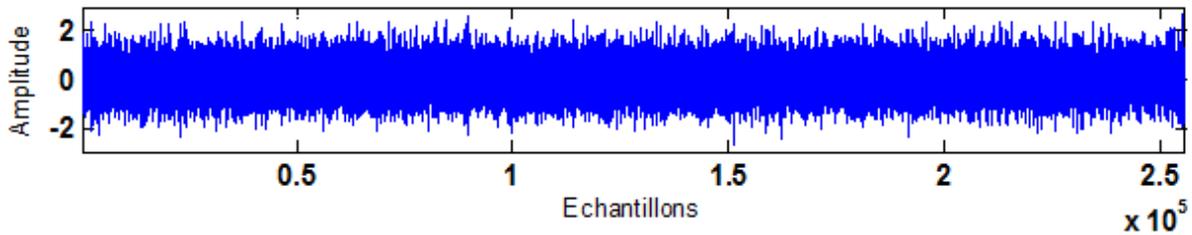


Figure 4-2: signal BBG1AR20.

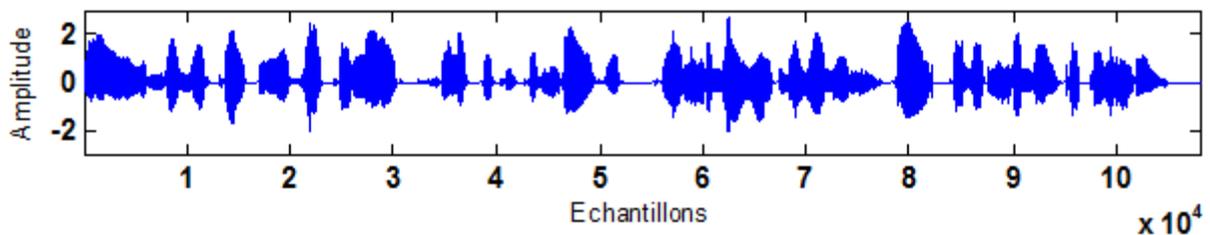


Figure 4-3: signal parole SPMX1.

Signal USASI (Figure 4.1) :

Le signal USASI (United States of America Standards Institute) est un bruit stationnaire réel à spectre moyen de la parole avec une dynamique spectrale de 32dB, ce bruit peut nous servir comme un signal de test dans des applications d'annulation d'écho acoustique ainsi pour l'évaluation de la vitesse de convergence des algorithmes adaptatifs.

Signal BBG1AR20 (Figure 4.2) :

C'est un bruit blanc Gaussien filtré avec un modèle autorégressif (filtre RII) d'ordre 20 qui est une réalisation d'un processus aléatoire stationnaire dans lequel la densité spectrale de

puissance possède une dynamique spectrale d'environ 42dB . C'est un signal plus corrélé que le signal USASI.

Signal Parole SPMX1 (Figure 4.3) :

C'est un signal parole réel obtenu par la concaténation de deux voix (femme +Homme), il possède une dynamique spectrale de 46 dB. Ce signal non stationnaire à long terme et peut être supposé stationnaire sur des périodes de 20 à 30ms.

Signaux d'Echo :

Ces signaux sont obtenus par convolution des signaux d'entrées par des réponses impulsionnelles acoustiques d'ordres 256 et 512 points données sur les figures (4.4) et (4.5). La réponse impulsionnelle mesurée dans l'habitacle d'une voiture est illustré ci-dessous.

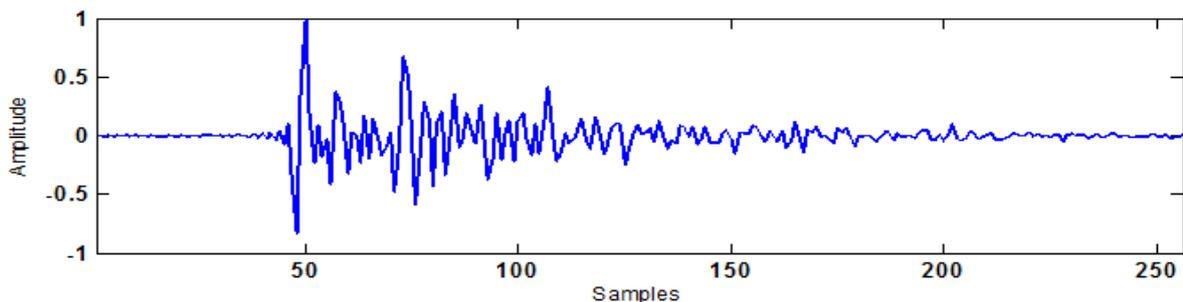


Figure 4-4: CAR impulse response (L=256).

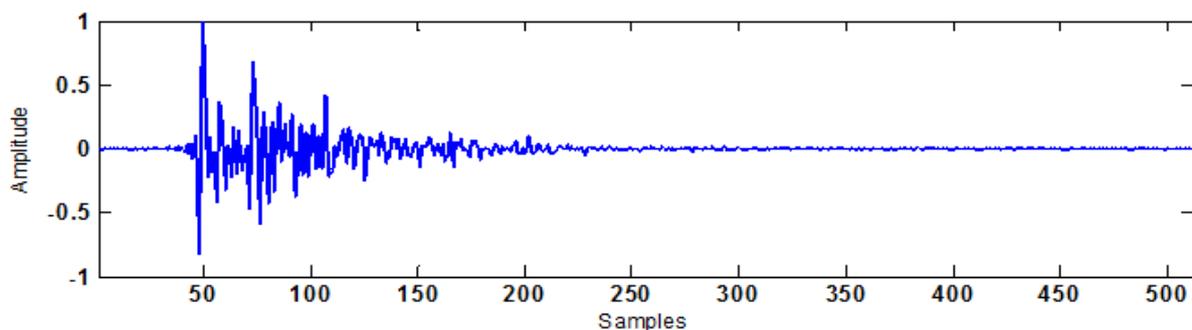


Figure 4-5: CAR impulse response (L=512).

Le bruitage du signal écho est réalisé par la fonction Matlab **awgn (y,SNR)**. Cette fonction ajoute au signal d'écho $y(n)$ un bruit blanc Gaussien avec un rapport signal sur bruit SNR en dB.

Les paramètres :

A propos des algorithmes OP-NLMS et OP-NLMS récursif avec blanchiment, nous avons utilisé les paramètres proposés dans le projet de fin d'étude [6], nous citons les valeurs initiales $\mathbf{m}_0 = \mathbf{0.1}$ et $\sigma_{z,0}^2 = \mathbf{0}$.

Nous avons aussi supposé σ_v^2 déjà connue, puis calculé sa valeur par la méthode théorique pour les deux algorithmes :

$$SNR = 10 \log_{10} \frac{\sigma_y^2}{\sigma_v^2} \quad (4.2)$$

$$\sigma_v^2 = 10^{-SNR/10} \cdot \sigma_y^2 \quad (4.3)$$

Où σ_y^2 désigne la puissance de l'écho.

Pour les tests, nous avons utilisé trois valeurs différentes pour le rapport signal sur bruit SNR=15, 30 et 50 dB. Deux tailles du canal acoustique (filtre FIR) ont été aussi utilisées dans nos simulations L=256 et 512.

Pour les estimateurs récursifs de la puissance du signal $\sigma_{x,n}^2$ et des prédicteurs d'ordre 1 et 2, nous avons respectivement deux facteurs d'oubli λ et λ_a . Dans la majorité des simulations, nous avons fixé $\lambda=0.992$ et $\lambda_a=0.96$. Ces valeurs peuvent changer selon le degré de non stationnarités du signal comme la parole.

La constante de régularisation c_a qui évite les divisions par des zéros surtout dans les zones de silence du signal parole est fixée à une valeur 0.01.

4.4 Simulations :

Pour voir le comportement de l'algorithme proposé par rapport à l'algorithme OP-NLMS dans le domaine de l'identification des réponses impulsionnelles acoustiques, nous avons réalisé plusieurs expériences avec trois types de signaux d'entrées, deux tailles de filtre et trois niveaux du SNR et ceci avec les trois types de prédiction décrits précédemment : prédiction LPC de Matlab, prédicteurs récurrents de premier et de deuxième ordre.

➤ Prédiction matlab d'ordre P :

Nous avons utilisé la routine Matlab `lpc` pour les signaux stationnaires USASI et BBG1AR20 pour un ordre de prédiction $P=1$ et des tailles du système à 256 et 512 et les trois niveaux du SNR. Les résultats sont donnés par les figures (4.6) à (4.13). Ces résultats, montrent clairement les améliorations en vitesse de convergence et en EQM finale de l'algorithme proposé (OPNLMS-BL, courbes en rouge) en comparaison avec l'algorithme OP-NLMS classique et ceci pour les deux types de signaux testés et quelques soient les tailles (256 et 512) et les niveaux de bruit SNR.

- **Signal USASI, P=1 :**

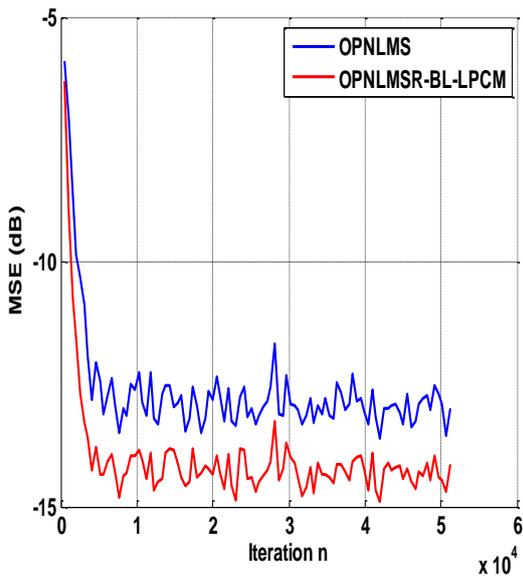


Figure 4-6: USASI, SNR=15dB, L=256 et P=1 .

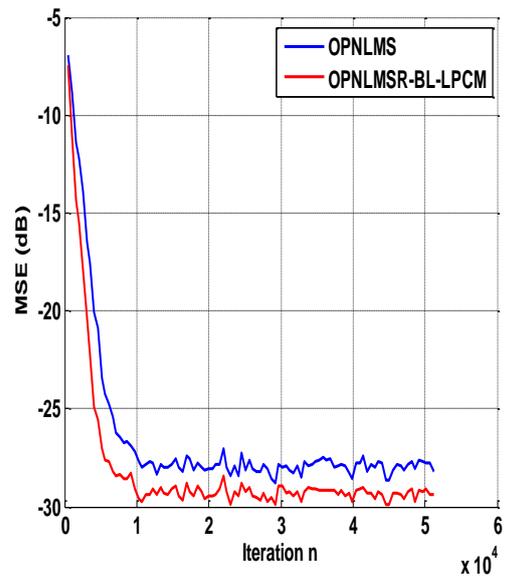


Figure 4-7: USASI, SNR=30dB, L=256 et P=1.

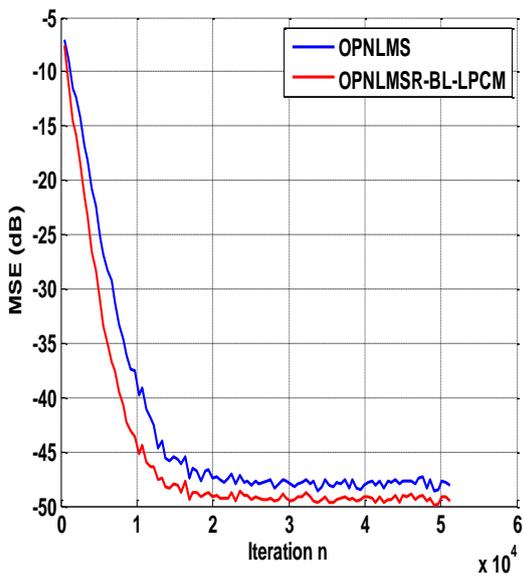


Figure 4-8: USASI, SNR=50dB, L=256 et P=1.

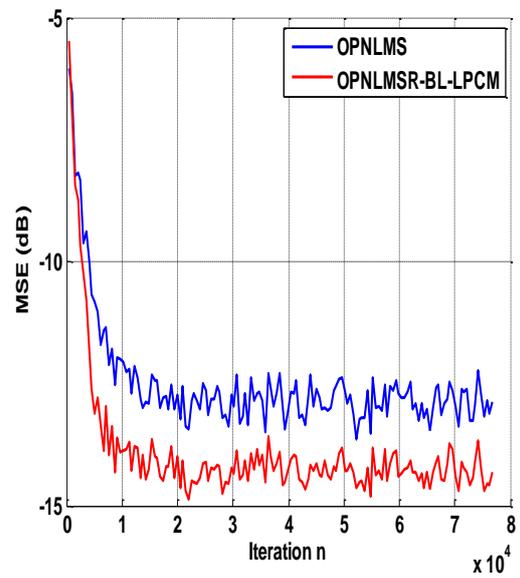


Figure 4-9: USASI, SNR=15dB, L=512 et P=1.

- Signal BBG1AR20, P=1 :

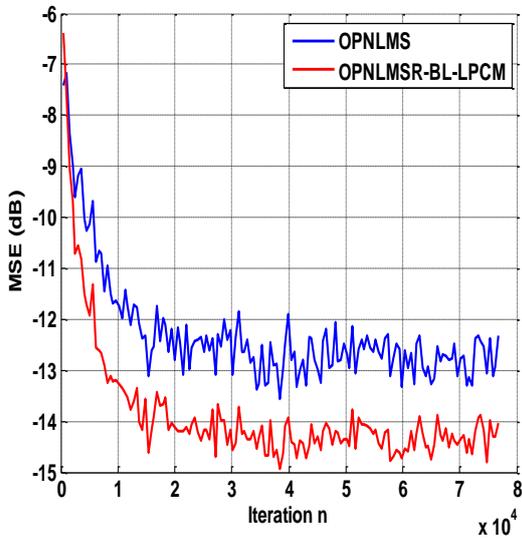


Figure 4-10: BBG1AR20, SNR=15dB, L=256 et P=1.

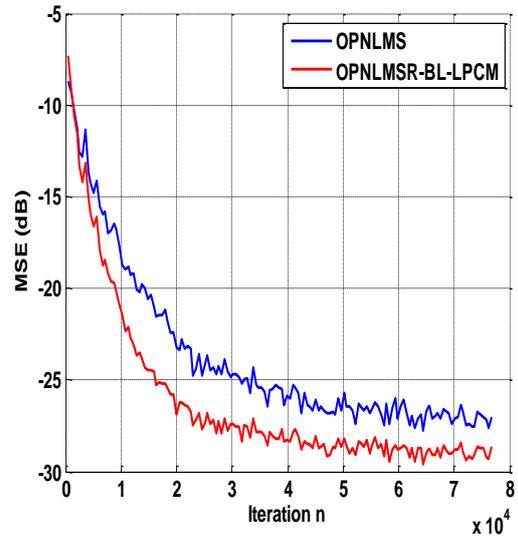


Figure 4-11: BBG1AR20, SNR=30dB, L=256,P=1.

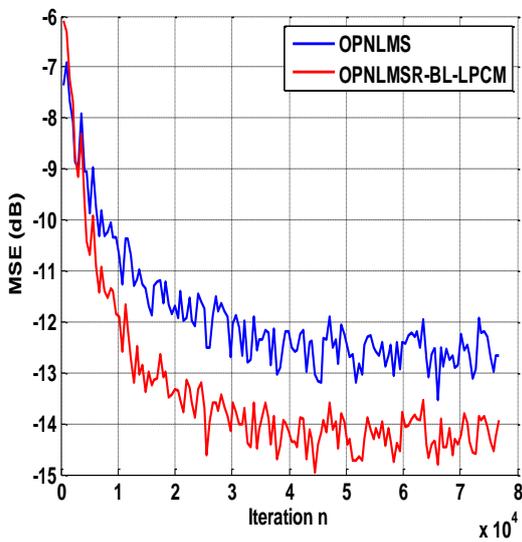


Figure 4-12: BBG1AR20, SNR=15dB, L=512 et P=1.

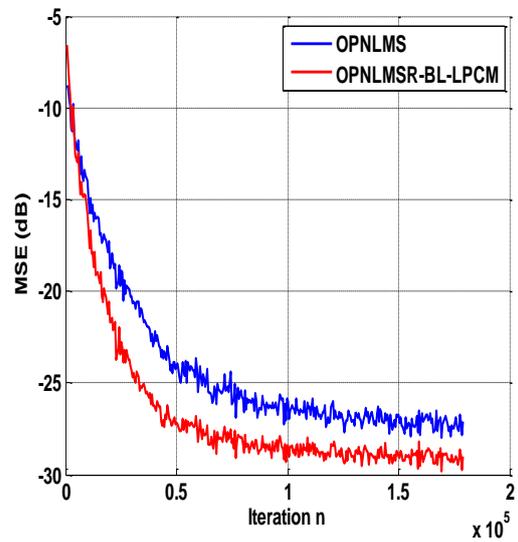


Figure 4-13: BBG1AR20, SNR=30dB, L=512 et P=1.

- **Comparaison P=1, 2 et 7 ou 10:**

Pour voir l'effet de la taille du prédicteur sur les performances de l'algorithme P-NLMS-BL, à l'aide de routine lpc de Matlab, nous avons simulé l'algorithme proposé avec un ordre de prédiction variable. Pour les mêmes conditions de simulation que le cas de P=1 précédent, les résultats de simulation sont donnés sur les figures (4.14) à (4.17).

L'observation évidente de ces résultats est qu'un ordre de prédiction $P > 1$ n'implique pas une amélioration de performance pour l'algorithme proposé. la meilleur performance est atteinte quand $P = 1$. Nous notons même une dégradation de performance pour le signal BBG1AR20 si l'ordre de prédiction vaut la valeur de 2 ou 7 et un SNR égale à 15dB (voir figure 4.17).

Cette observation pour $P = 1$ sera confirmée par la suite en comparant les prédicteurs récursifs d'ordre 1 et 2.

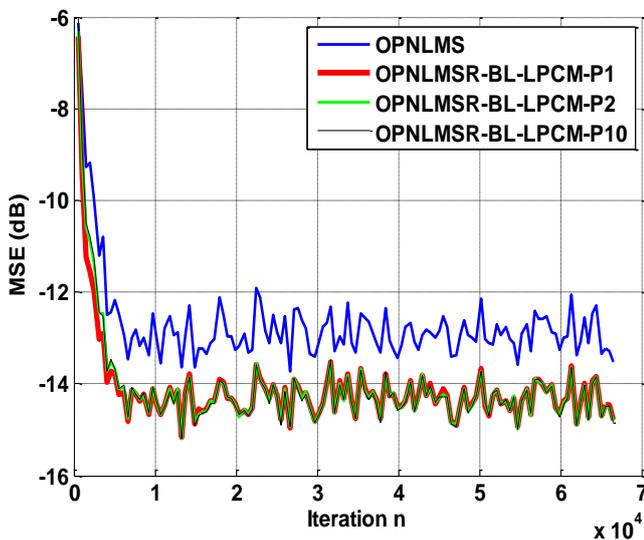


Figure 4-14: USASI, SNR=15dB,
L=256 et P=1,2 et 10 .

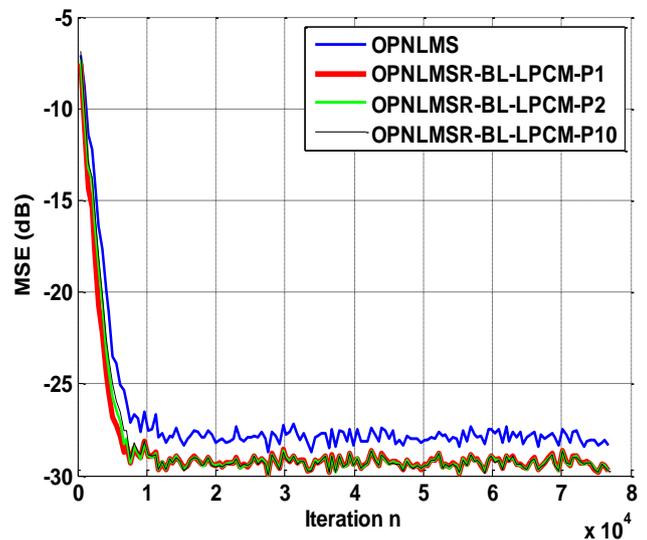


Figure 4-15: USASI, SNR=30dB,
L=256 et P=1, 2 et 10.

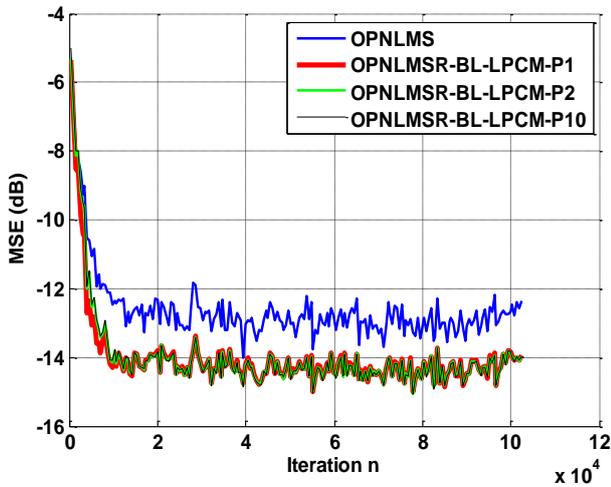


Figure 4-16: USASI, SNR=15dB, L=512 et P=1,2 et 10.

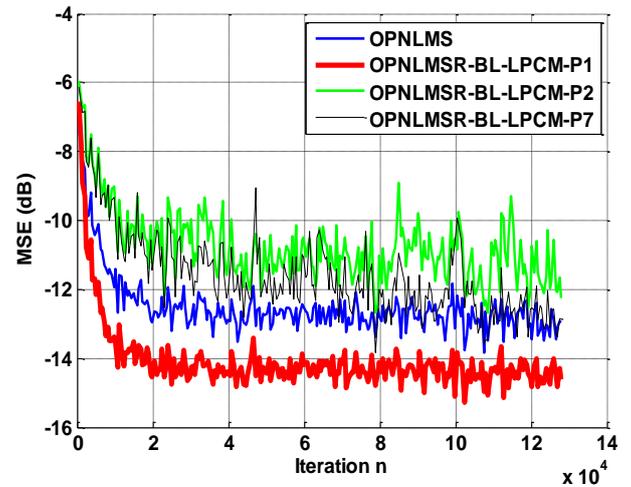


Figure 4-17: BBG1AR20, SNR=15dB, L=256 et P=1,2 et 7.

➤ Tests avec prédicteur récursif de premier ordre:

Nous allons refaire les mêmes tests pour le prédicteur récursif d'ordre 1 pour les différents signaux et tailles du filtre ainsi pour les trois SNR considérés. Par la suite nous comparons le prédicteur d'ordre 1 et 2 récursifs.

Les mêmes séries de simulations ont été effectuées pour l'algorithme proposé avec P=1 récursif. Une sélection de résultats est donnée sur les figures (4.18) à (4.25). Nous observons que les performances de l'algorithme proposé sont supérieures aux performances de l'algorithme OP-NLMS classique, et ceci pour les deux signaux stationnaires considérés, et pour les deux tailles du filtre et les trois valeurs du SNR. Nous notons que ces améliorations deviennent plus faibles pour des valeurs du SNR importants (signaux moins bruités). Ceci est logique car l'algorithme OP-NLMS a été proposé pour améliorer les performances en milieu très bruité. En absence de bruit, les algorithmes NLMS et OP-NLMS deviennent similaires.

- Signal USASI, P=1 récursif :

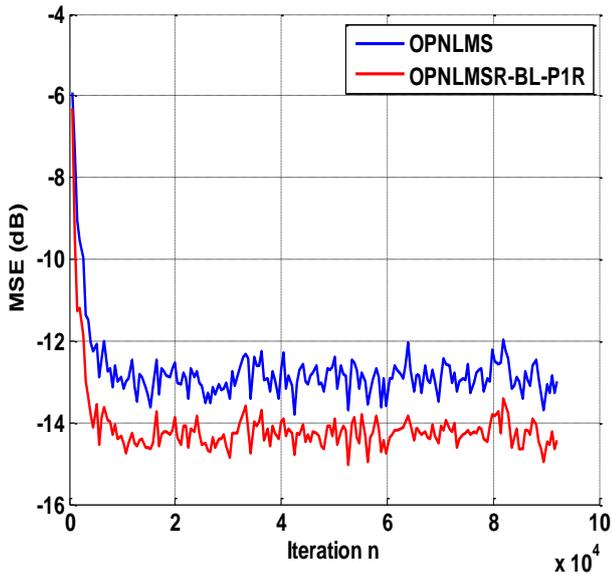


Figure 4-18: USASI, SNR=15dB, L=256 et P=1 récursif.

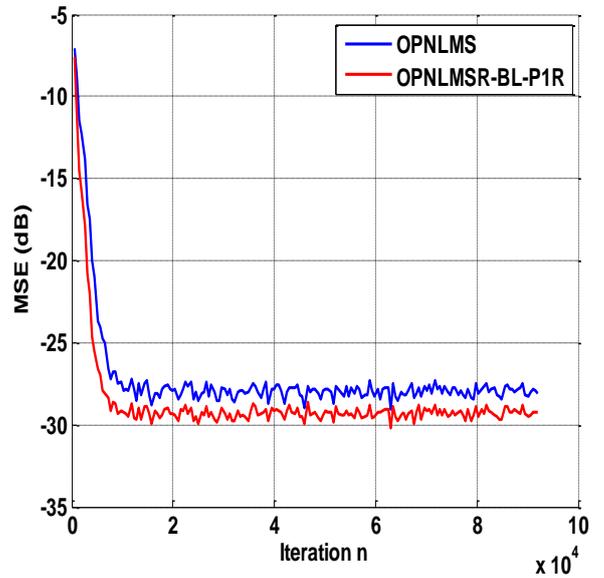


Figure 4-19: USASI, SNR=30dB, L=256 et P=1 récursif.

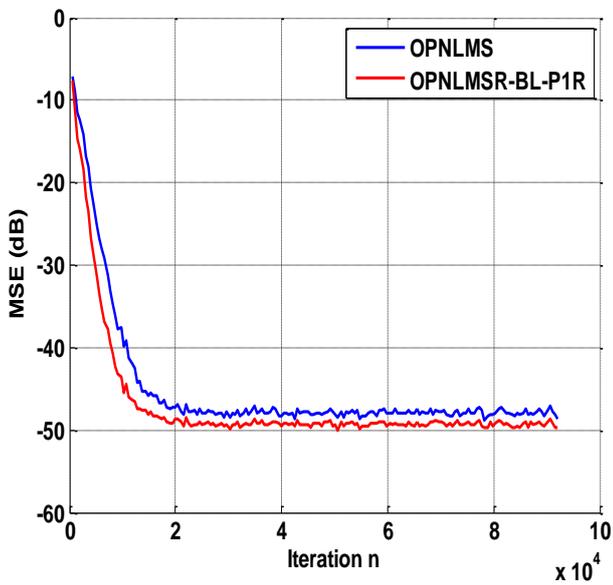


Figure 4-20: USASI, SNR=50dB, L=256 et P=1 récursif.

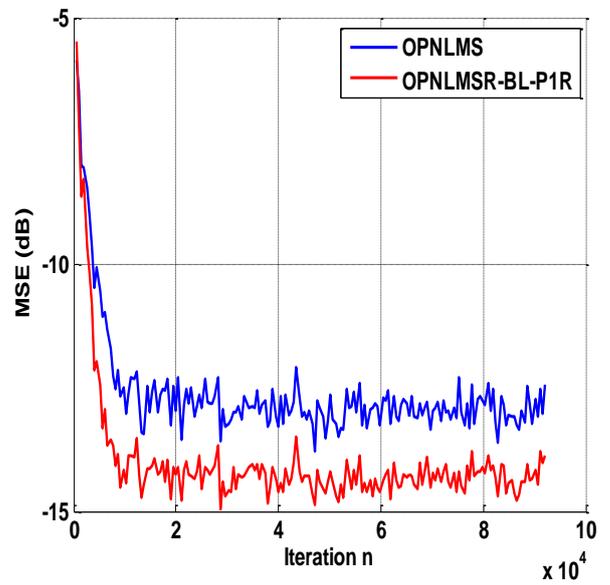


Figure 4-21: USASI, SNR=15dB, L=512 et P=1 récursif.

- Signal BBG1AR20, P=1 récurrent:

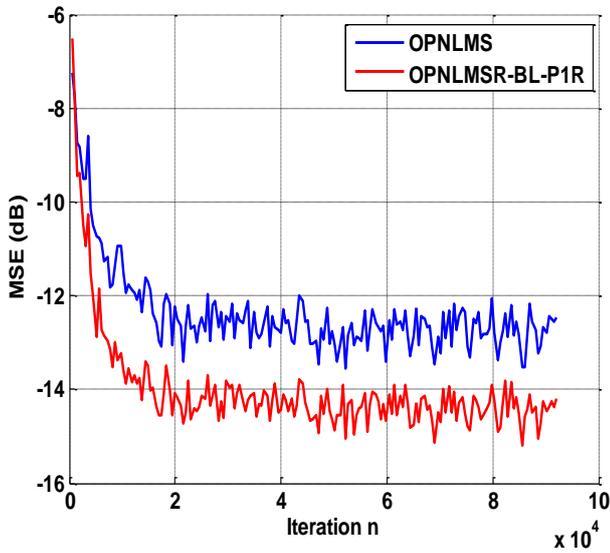


Figure 4-22: BBG1AR20, SNR=15dB, L=256 et P=1 récurrent .

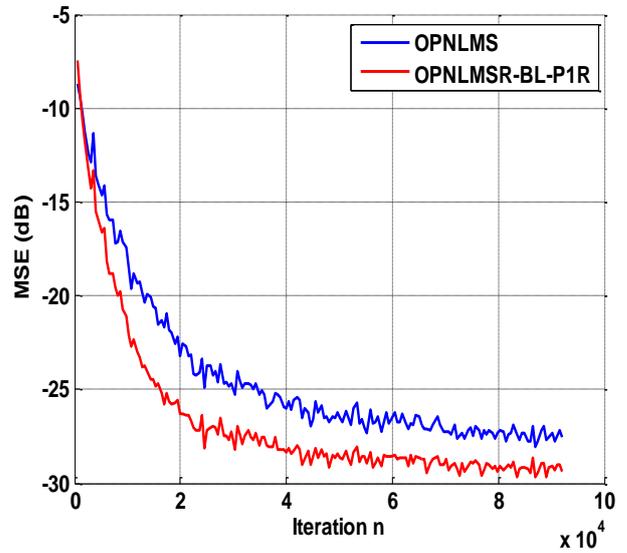


Figure 4-23: BBG1AR20, SNR=30dB, L=256 et P=1 récurrent.

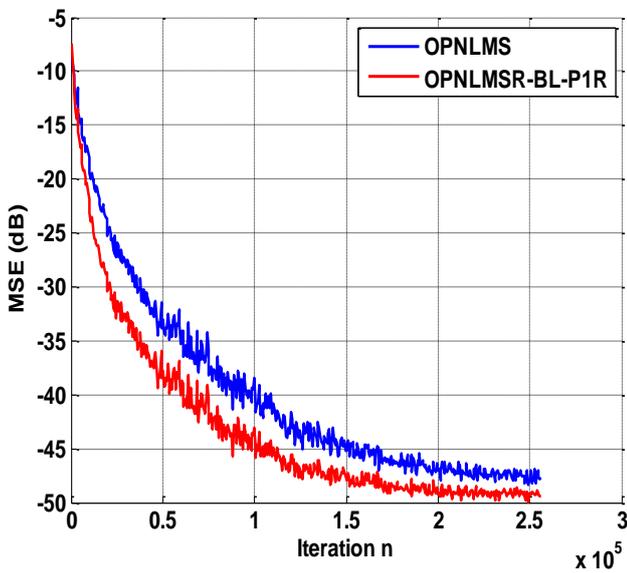


Figure 4-24: BBG1AR20, SNR=50dB, L=256 et P=1 récurrent.

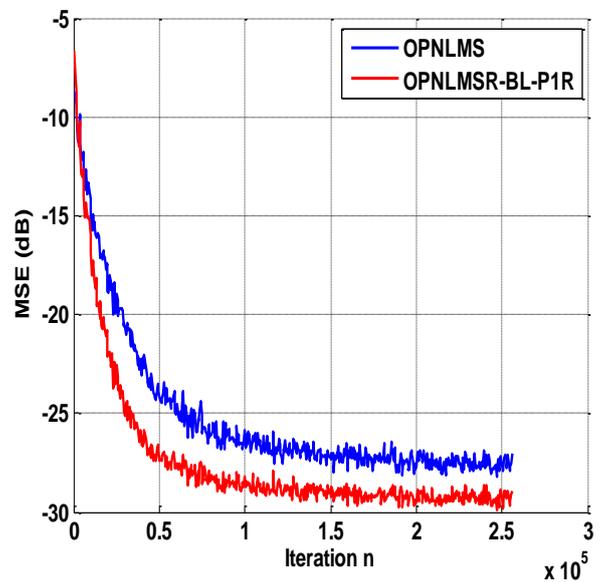


Figure 4-25: BBG1AR20, SNR=30dB, L=512 et P=1 récurrent.

➤ Comparaison des prédicteurs P1 et P2 récursifs :

Les figures (4.26) à (4.29), comparent les performances des prédicteurs récursifs de premier et de deuxième ordre ,et ont confirmé les résultats trouvés avec la routine lpc de Matlab . les performances de ces prédicteurs sont presque identiques, d'autre part nous notons un léger avantage pour le prédicteur à premier ordre en EQM finale.

Par la suite, nous allons tester l'algorithme proposé avec le prédicteur de premier ordre en utilisant un signal non stationnaire comme la parole.

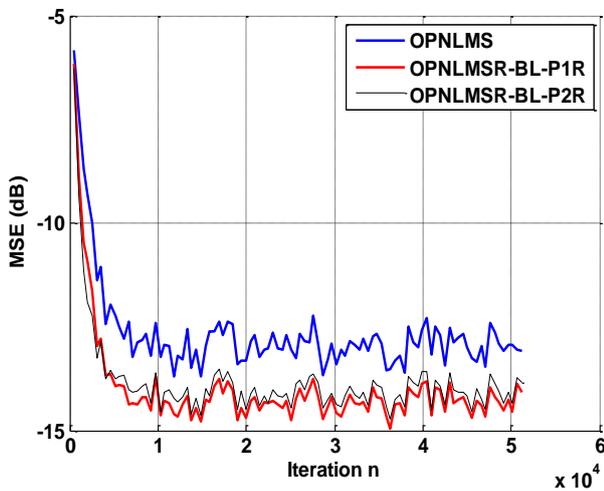


Figure 4-26: USASI, SNR=15dB,
L=256 et P=1, 2 récursifs.

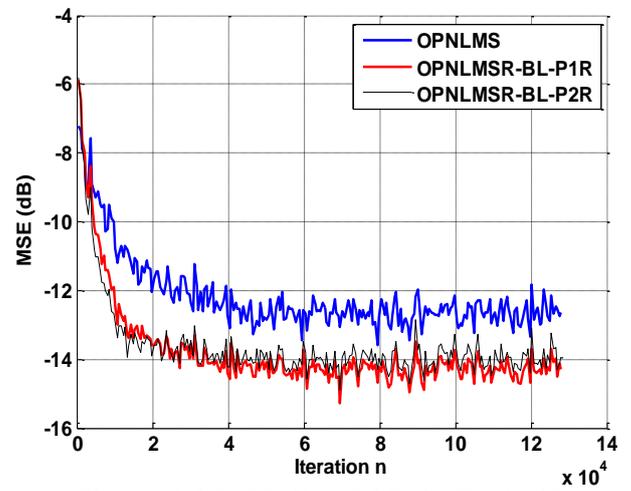


Figure 4-27: BGG1AR20, SNR=15dB,
L=512 et P=1, 2 récursifs.

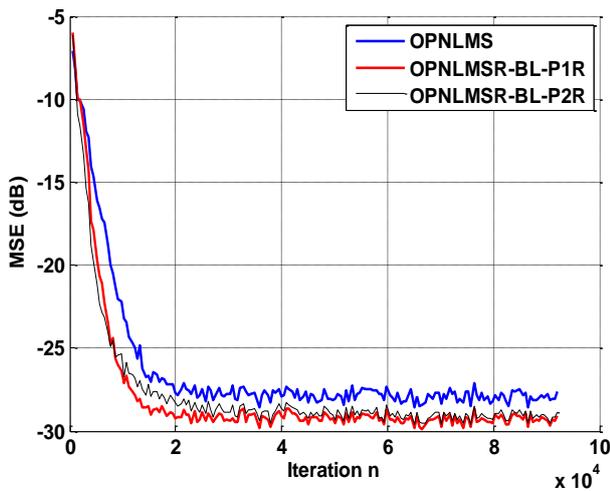


Figure 4-28: USASI, SNR=30dB,
L=512 et P=1,2 récursifs.

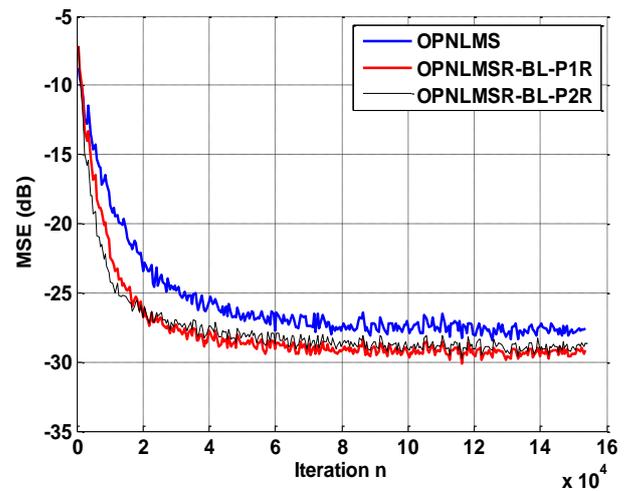


Figure 4-29: BGG1AR20, SNR=30dB,
L=256 et P=1,2 récursifs.

➤ Tests avec le signal de parole SPMX1 :

Le signal parole est un signal fortement non stationnaire avec une dynamique spectrale importante, contenant des zones de silences (absence de signal), ce qui engendre des difficultés dans l'identification pour les algorithmes de type NLMS.

Par ailleurs, comme il a été montré dans les paragraphes précédents quand l'ordre de prédiction $P > 1$ aucune amélioration de performances n'est apporté dans l'algorithme proposé. Nous allons alors tester cet algorithme à l'aide du prédicteur d'ordre 1 récursif avec un signal non stationnaire (la parole).

Les résultats de simulation réalisés avec la parole pour différentes tailles du filtre et les trois valeurs du SNR considérés sont donnés sur les figures (4.30) à (4.35). D'après ces résultats, nous notons les difficultés de l'algorithme OP-NLMS (courbe en bleu) pour atténuer l'écho surtout pour les cas bruités (SNR=15 et 30dB). Par contre, l'algorithme proposé avec un prédicteur d'ordre 1 montre des performances très intéressantes avec la parole (courbes en rouge) ; le niveau de l'écho atténué est proche du niveau de bruit considéré. Ce qui rend l'algorithme proposé avec un prédicteur d'ordre 1 récursif un très bon candidat pour les applications d'annulation d'écho acoustique en milieu bruité.

✓ $L=256$:

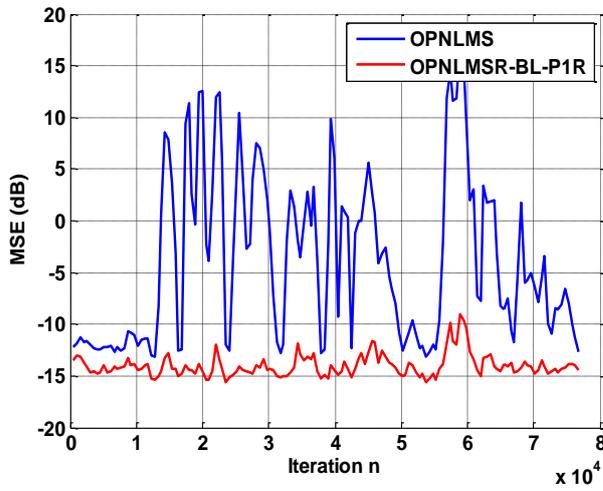


Figure 4-30: Parole SPMX1, SNR=15dB,
 $L=256$, $P=1$ R, $\lambda=0.9986$.

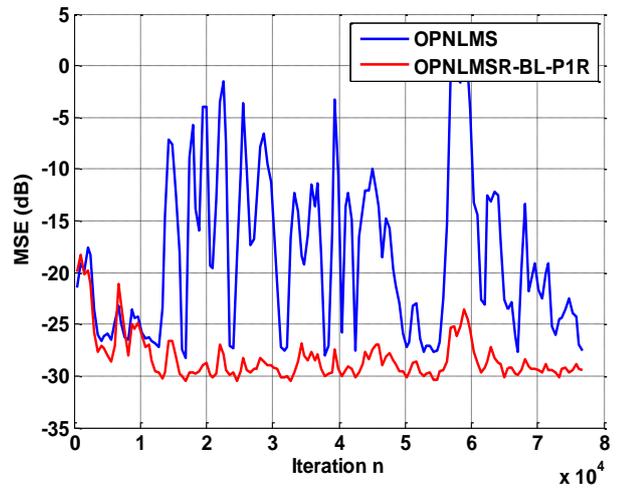


Figure 4-31: Parole SPMX1, SNR=15dB,
 $L=256$, $P=1$ R, $\lambda=0.9986$.

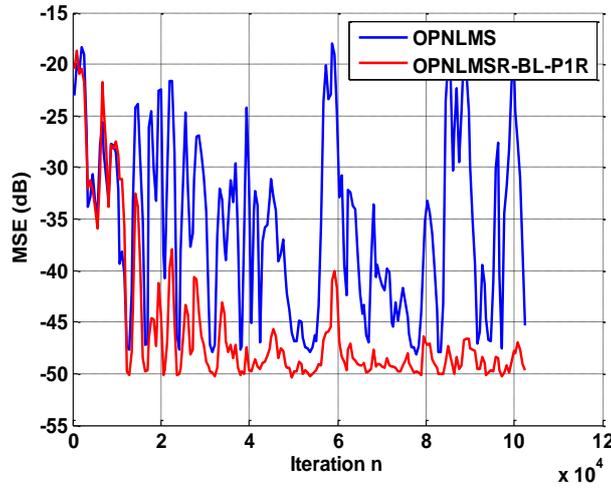


Figure 4-32: Parole SPMX1, SNR=50dB,
 $L=256$, $P=1$ R, $\lambda=0.9986$.

✓ $L=512$:

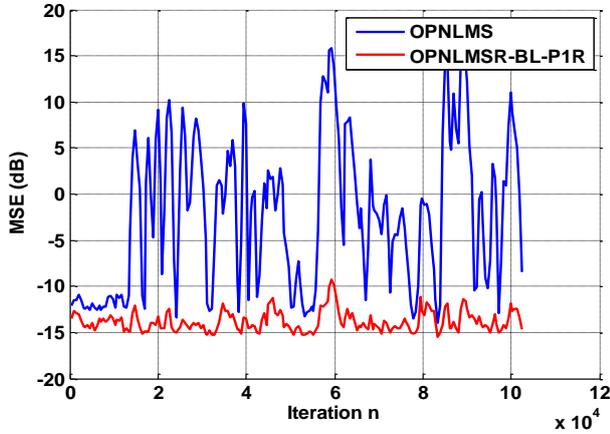


Figure 4-33: Parole SPMX1, SNR=15dB, L=512, P=1 R, $\lambda=0.9986$.

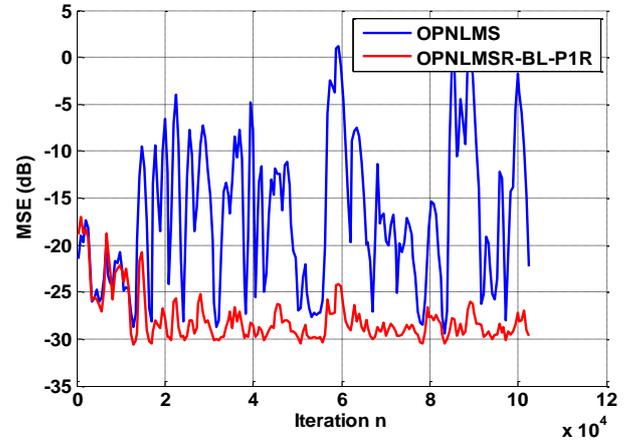


Figure 4-34: Parole SPMX1, SNR=30dB, L=512, P=1 R, $\lambda=0.9986$.

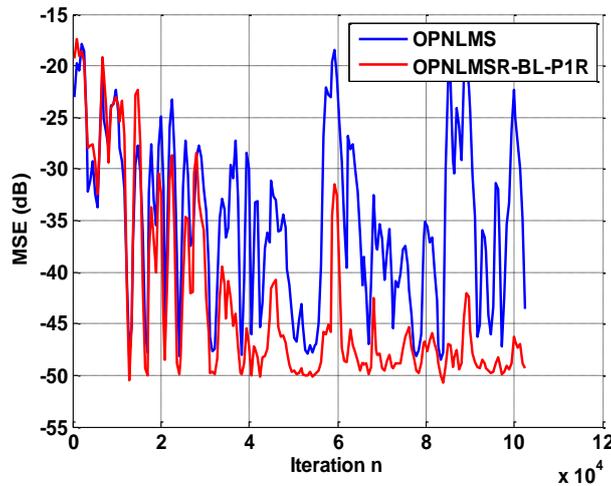


Figure 4-35: Parole SPMX1, SNR=50dB, L=512, P=1 R, $\lambda=0.9986$.

4.5 Conclusion :

Dans ce chapitre, nous avons pu présenter les résultats de simulations de l'application des algorithmes de filtrage adaptatif OP-NLMS et OP-NLMS-BL sur les signaux stationnaires bbg1AR20, USASI et le signal non-stationnaire SPMX1 (parole).

Les résultats de simulation ont montré le bon comportement de l'algorithme OP-NLMS-BL en comparaison avec l'algorithme OP-NLMS classique.

Les meilleures performances, en vitesse de convergence et EQM finale, ont été observées avec l'algorithme proposé avec un prédicteur d'ordre 1, ce qui rend cet algorithme moins complexe en opérations arithmétiques.

L'algorithme proposé avec P1 récursif, montre aussi la robustesse et les performances très intéressantes avec la parole. Ce qui fait de l'algorithme un très bon candidat pour les applications d'annulation d'écho acoustique en milieu bruité.

Conclusion générale

Conclusion générale :

L'emploi d'un dispositif de communication main libre est accompagné de plusieurs problèmes critiques qui affaiblissent l'intelligibilité de la parole transmise tel que l'écho acoustique .à cet effet l'évolution des télécoms mobiles grand public suscite l'annulation de l'écho acoustique par des algorithmes adaptatives.

Dans le cadre de ce mémoire les travaux élaborés fournissent une solution prêtant pour éliminer l'écho acoustique .

Dans un premier temps nous avons apporté une généralité sur l'écho acoustique et son annulation, ensuite nous avons exposé quelques notions sur le filtrage adaptatif ainsi que sur les familles des algorithmes adaptatives en prêtant une grande intention à la famille des gradients stochastiques en raison de leur vitesse de convergence qui dépend de la dynamique spectrale, sachant qu'une lente vitesse de converge est du à une dynamique spectrale très importante . de ce fait nous avons conçu une technique de blanchiment qui aplatit la dynamique spectrale pour offrir une vitesse de convergence rapide .

Après avoir comparer à l'aide des simulations matlab , en utilisant la prédiction linéaire (LPC) matlab ainsi que les prédicteurs récursifs d'ordre 1 et 2, plus l'effet du SNR et de l'ordre de prédiction , nous avons constaté qu'un prédicteur de premier ordre est un excellent choix permettant d'accélérer les opérations arithmétiques et de progresser la robustesse notamment les performances de notre algorithmes , des résultats prometteurs sont également attribué dans le cas d'un signal parole .

En tous plusieurs recherches sont encore ouvertes .nous espérant que le travail accompli à fournit des interprétations suffisantes sur les résultats obtenus et qu'il ouvrira plusieurs perspectives pour les futurs travaux . nous envisageons également d'élargir notre étude sur

l'algorithme "OP-NLMS-BL" pour mettre en valeurs d'autres facteurs importants dans l'annulations d'écho .

Bibliographie

- [1] :Fatiha Taihi , Samia Benmaiza ‘‘Etude et implémentation d’un filtre RIF adaptatif sous VHDL ‘‘ , mémoire de master électronique des Systems embarqués , département d’électronique ,Université Blida 1 , 2018 .
- [2] : Majid Arezki ‘‘algorithmes adaptatifs évolués de filtrage ‘‘ , thèse de doctorat en électronique, école nationale polytechnique Alger, 2011.
- [3] : Khaled Ferrane, Ibtissem Sadouni ‘‘filtrage adaptatif pour l’annulation d’écho acoustique dans les communications à mains libres ‘‘ , mémoire de master, Faculté des sciences et des sciences appliquées, Université Ouargla ,2017 .
- [4] : Mohamed Ramdane, Abdelhak Kedjar ‘‘ algorithmes rapides pour l’identification de Systems parcimonieux ‘‘ , mémoire de master Filière Électronique Spécialité Système de télécommunication, département d’électronique, Université Blida 1 , 2018 .
- [5] : Réda Bendoumia ‘‘annulation du bruit par les méthodes deséparation de Sources aveugles’’ , thèse de doctorat en Génie Electrique, département d’électronique, Université Blida1, 2014.
- [6] : Nadjat Bouzertini, Kheira Seraan ‘‘Algorithme NLMS optimisé pour l’identification de réponses impulsionnelles acoustique ‘‘ , mémoire de master Spécialité Système de télécommunication, département d’électronique , Université Blida 1 , 2019.
- [7] :Mohamed Najim ‘‘Filtrage optimal ‘‘ Ecole Nationale Supérieure d’Electronique et de Radioélectricité de Bordeaux (ENSERB) (1998).
- [8] :Ahmed Benallal, Abdelhak Benkrid "A simplified FTF-Type algorithm for Adaptive Filtering" Signal Processing Journal (Elsevier) 87 (2007).
- [9] : Silviu Ciochină , Constantin Paleologu , Jacob Benesty ‘‘An optimized NLMS algorithm for system identification ‘‘ , signal processing Journal (2016).
- [10] :Constantin Paleologu, Silviu Ciochin, Jacob Benesty and Steven L ‘‘ An overview on optimized NLMS algorithms for acoustic echo cancellation ‘‘ , Paleologu *et al. EURASIP Journal on Advances in Signal* (2015).
- [11] :Ahmed Benallal, Majid Arezki ‘‘nouvel algorithme de type NLMS à convergence rapide pour le filtrage adaptatif ‘‘ XXIVème Colloque GRETSI, 3-6 sept 2013, Brest - France
- [12] :Mamadou Mboup , Madeleine Bonnet ‘‘ une nouvelle structure blanchissante pour l’annulation de l’écho acoustique ‘‘ 13eme colloque JAN-LES-PINS GRETSI(1991) France.
- [13] : J.K Soh et SC. Douglas ‘‘ l’implémentation de l’algorithme NLMS avec des filtres de décorrelation ‘‘ Proc. Int. Workshop on Acoustic Echo and Noise Control, Sept. 1999

[14] : Sofia Ben Jebara ,Hichem Besbes ‘‘Une famille d’algorithmes de signe à base de structures blanchissantes’’ Colloque International Telecom 2005, Rabat, Morocco, March 2005.

[15] :Jonat Han Huang , Kuan Chieh Yen , Yunxin Zhao ‘‘un filtrage adaptatif de décorrelation basé sur les sous bandes ‘‘IEEE Transactions on Speech and Audio Processing 2000.

[16] : S. Gazor and T. Liu ‘‘Adaptive filtering with decorrelation for coloured AR environments’’ IEE Proceedings 2004.