

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

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

جامعة سعد دحلب البلدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Mention Électronique
Spécialité Automatique

présenté par

BOUGUERRA Islam

&

KHALFAOUI Ismail

Identification paramétrique basée sur la prédiction et application au signal de parole

Proposé par : Mr. MADDI Abdelkader

Année Universitaire 2016-2017

Remerciements

Elssalam alikom.

Avant de présenter notre travail Nous tenons à exprimer nos vifs remerciements à notre promoteur **Mr : AEK MADDI** pour nous avoir encadrés durant notre projet de fin d'étude et nous conseillé tout le long de notre travail.

Nous remercions également tous les enseignants qui nous ont aidés à surmonter les problèmes que nous avons rencontrés dans le processus que nous recherchons et tout qui nous ont aidés à la réalisation de projet.

Nous remercions les membres de jury qui ont fait l'honneur d'examiner notre travail

Dédicace

Je dédie ce travail à
Mes très chers parents,
Mes très chers frères et sœurs
Mon binôme qui ma supporter tous ces années,
Toutes mes amis et mes collègues d'études,
Mes professeurs dans toutes les pistes d'études,

ISLEM

Je dédie ce travail à
Mes très chers parents,
Mes très chers frères et sœurs
En particuliers ma grand mère
Mon binôme qui ma supporter tous ces années,
Toutes mes amis et mes collègues d'études,
Et à tous ceux qui me sont chers.

SMAIL

الخلاصة:

الهدف من هذا العمل هو دراسة وتنفيذ خوارزميات تحديد الأبعاد القائمة على تبييض الخطأ للحصول على خطأ يؤول إلى الضوضاء البيضاء. وسنطبق هذه الخوارزميات المطبقة على النماذج (ARX, ARMAX, BJ , OE). وأخيراً، نقترح تطبيق عملي لهذه الطريقة وهي إشارة الكلام أو الصوت الإنساني الذي سيكون ثمرة العمل الذي نقوم به.

كلمات المفاتيح: تحديد الأبعاد. الضوضاء البيضاء. الصوت الإنساني.

Résumé : Le but de ce projet de fin d'étude, consiste à étudier et implémenter sous MATALAB, la technique d'identification paramétrique basée sur la prédiction, où nous supposerons que le modèle obtenu est un prédicteur, c'est à dire qu'il permet de calculer la sortie à l'instant présent en fonction des entrées et des sorties réelles aux instants précédents. La méthode utilisée est celle de la méthode des moindres carrés récursive appliquée aux modèles stochastiques ARX, ARMAX, BJ et OE, basés sur les données d'observation. Elle consiste à déterminer en premier lieu, d'une façon récursive, un prédicteur linéaire, pour identifier les paramètres du système physique basée sur le blanchissement de l'erreur de prédiction. Ensuite, pour tester les performances de la méthode proposée, plusieurs testes de simulations doivent être effectués sur MATLAB. Enfin, nous proposons une application pratique de cette méthode au signal de parole.

Mots clés : Identification ; Prédiction ; Modèle ARMAX ; Erreur de sortie ; Parole.

Abstract: The aim of this work is to study and implement the parametric identification technique based on prediction error, where we assume that the model obtained is a predictor; it makes it possible to calculate the output at the present time as a function of the actual inputs and outputs at the previous instants. The method used here is the recursive least squares method applied to the ARX, ARMAX, BJ and OE stochastic models based on the observation data. It consists in first determining, in a recursive way, a linear predictor, to identify the parameters of the physical system based on the whitening prediction error. Then, to test the performance of the proposed method, several simulation tests must be carried out on MATLAB. Finally, we propose a practical application of this method to the speech signal.

Keywords: Identification; Prediction; ARMAX model; Output error; Word.

Listes des acronymes et abréviations

ARMA	Auto Régressive à Moyenne Ajustée
ARMAX	Auto Régressive à Moyenne Ajustée avec entrée exogène
AR	Auto Régressif
MA	Moyenne Ajustée
OE	Output Error
BJ	Box-Jenkins
SBPA	La Séquence Binaire Pseudo Aléatoire
RLS	Recursive Least Square
RELS	Recursive Extended Least Square
MCE	Moindres Carrées Etendus
MCR	Moindres Carrées Récursif
SNR	Signal to Noise Ratio (rapport signal sur bruit)
t	temps (discret)

Table des matières

Remerciement

Dédicace

Résumé

Listes des acronymes et abréviation

Liste des figures

Liste des tableaux

Introduction générale 1

Chapitre 1 :

Modélisation et identification

1.1 Introduction..... 3

1.2 Modélisation..... 3

1.2.1 Définition 3

1.2.2 But de la modélisation..... 3

1.2.3 Modèles mathématiques 4

a) *Définition* 4

b) *Intérêts*..... 4

c) *Conception* 4

1.2.4 Principaux étapes de la modélisation..... 5

a) *Système*..... 5

b) *Modèle*..... 5

c) *Critère* 6

d) *Optimiseur* 7

e) *Incertitude sur les paramètres*..... 7

f) *Analyse critique des résultats obtenus* 7

1.2.5 Modélisation autorégressive à moyenne ajustée (ARMA)..... 7

1.2.6	Modèle ARMAX	8
1.2.7	Modèle à erreur de sortie (OE)	9
1.2.8	Modèle Box-Jenkins (bj)	9
1.3	Identification	10
1.3.1	Définition	10
	<i>a) Identification non paramétrique</i>	10
	<i>b) Identification paramétrique</i>	10
1.3.2	Les étapes d'identification	10
	<i>a) Acquisition des entrées-sorties sous un protocole d'expérimentation</i>	11
	<i>b) Choix ou estimation de la complexité du modèle</i>	11
	<i>c) Estimation des paramètres du modèle</i>	11
	<i>d) Validation du modèle</i>	11
1.3.3	Algorithme d'identification récursif	12
1.4	Conclusion	12

Chapitre 2 :

L'algorithme des moindres carrés

2.1	Introduction	14
2.2	Définition de la méthode des moindres carrés	14
2.2.1	Identification du modèle ARMA	15
2.2.2	Solution optimale au sens des moindres carrés	16
2.2.3	Algorithme des moindres carrés récursifs (RLS)	17
2.2.4	Algorithme des moindres carrés récursif pondéré λ RLS	18
2.2.5	Propriétés de l'estimateur des moindres carrés	19
	a) Définition du biais	19
	b) Absence de biais	20
2.3	Etude et développement de la méthode des moindres carrés étendus	20

2.3.1	Introduction.....	20
2.3.2	Algorithme des moindres carrés étendus	21
2.3.3	Détermination des paramètres	22
2.3.4	Equations de la mise en œuvre de l'algorithme RELS.....	23
2.3.5	Initialisation de l'algorithme RELS.....	24
2.4	Algorithme des moindres carrés avec modèle OE.....	24
2.4.1	Le modèle avec erreur de sortie (OE)	24
2.4.2	Le prédicteur d'un modèle OE	24
2.4.3	La forme matricielle	25
2.5	Algorithme des moindres carrés avec modèle Box-Jenkins.....	25
2.5.1	Le modèle Box-Jenkins	25
2.5.2	Le prédicteur d'un modèle BJ.....	26
2.5.3	La forme matricielle	26
2.6	Conclusion.....	27

Chapitre 3:

Simulation et interprétation des résultats

3.1	Introduction.....	28
3.2	Bruit blanc	28
3.3	Séquence d'entrée SBPA (Séquence Binaire Pseudo Aléatoire)	29
3.4	Algorithme des moindres carrés récursif (RLS)	29
3.4.1	Les processus de la simulation	29
3.4.2	Influence du nombre d'échantillons	30
3.4.3	Influence de la variance δ^2	35
3.4.4	Interprétation	37
3.5	Algorithmes des moindres carrés pondérés (λ RLS).....	37
3.5.1	Influence du facteur d'oubli λ	37

3.5.2	Interprétation	40
3.6	Algorithme des moindres carrés étendus (RELS)	41
3.6.1	Les processus de la simulation	41
3.6.2	Influence du nombre d'échantillons	42
3.6.3	Influence de la variance δ^2	46
3.6.4	Interprétation	47
3.7	Algorithme de moindres carrée avec modèle OE (Output Error)	48
3.7.1	Les processus de la simulation	48
3.7.2	Influence du nombre d'échantillons	48
3.7.3	Influence de la variance.....	50
3.7.4	Interprétation	51
3.8	Algorithme de moindres carrée avec modèle BJ (Box-Jenkins)	51
3.8.1	Les processus de la simulation	51
3.8.2	Influence de nombre d'échantillons.....	52
3.8.3	Influence de la variance.....	53
3.8.4	Interprétation	54
3.9	Conclusion	54

Chapitre 4:

Application au signal de parole

4.1	Introduction.....	55
4.2	Le mécanisme de phonation	55
4.3	Caractéristiques du signal de parole	57
4.3.1	Les sons voisés.....	57
4.3.2	Les sons non voisés.....	57
4.4	Analyse et paramétrisation de la parole	57
4.5	Résultat de simulations	58

a)	Partie voisée du signal de parole.....	59
b)	Partie non voisée du signal de parole(RLS)	61
c)	Résultat de simulations par l’algorithme RELS.....	63
4.6	Conclusion	64
	Conclusion générale.....	65
	Bibliographie.....	67

Liste des figures

<i>Figure 1.1.</i> Representation d'un systeme.....	5
<i>Figure 1.2.</i> Modèle parallèle.	6
<i>Figure 1.3.</i> Modèle série.	6
<i>Figure 1.4.</i> Erreur de sortie pour un modèle parallèle.	6
<i>Figure 1.5.</i> Exemple de circulation de l'information pour l'optimisation.....	7
<i>Figure 2.1.</i> Structure de réalisation d'un modèle ARMA.....	16
<i>Figure 2.2.</i> Structure de réalisation d'un modèle ARMAX.	21
<i>Figure 3.1.</i> Signal du bruit $e(t)$ et sa fonction d'auto-corrélation.	28
<i>Figure 3.2.</i> Séquence SBPA en haut et sa fonction d'auto-corrélation en bas	29
<i>Figure 3.3.</i> L'évolution des paramètres ($N = 256*4, \delta^2 = 0.0001$).....	31
<i>Figure 3.4.</i> Erreur de prédiction et sa fonction d'auto-corrélation.	31
<i>Figure 3.5.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001$).....	32
<i>Figure 3.6.</i> Le signal de l'erreur de prédiction et son auto corrélation.....	32
<i>Figure 3.7.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001$).....	33
<i>Figure 3.8.</i> Signal de l'erreur de prédiction.....	33
<i>Figure 3.9.</i> Fonction d'auto-corrélation de l'erreur.....	33
<i>Figure 3.10.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001$)	34
<i>Figure 3.11.</i> Erreur de prédiction.	34
<i>Figure 3.12.</i> Fonction d'auto-corrélation de l'erreur.....	35
<i>Figure 3.13.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001, \lambda =0.98$)	38
<i>Figure 3.14.</i> Evolution des paramètres ($N = 1024, \delta^2 = 0.0001, \lambda = 0.98$).....	38
<i>Figure 3.15.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001, \lambda =0.98$)	39
<i>Figure 3.16.</i> Evolution des paramètres ($N=1024, \delta^2=0.0001, \lambda =0.98$)	40
<i>Figure 3.17.</i> Evolution des paramètres ($N = 1024, \delta^2 = 0.001$).....	42
<i>Figure 3.18.</i> Auto-corrélation de l'erreur de prédiction.....	43

<i>Figure 3.19.</i> Evolution des paramètres (N = 1024, $\delta^2 = 0.001$).....	43
<i>Figure 3.20.</i> Auto-corrélation de l'erreur de prédiction.....	44
<i>Figure 3.21.</i> Evolution des paramètres (N = 1024, $\delta^2 = 0.001$).....	44
<i>Figure 3.22.</i> Auto-corrélation de l'erreur de prédiction.....	45
<i>Figure 3.23.</i> Evolution des paramètres (N = 1024, $\delta^2 = 0.001$).....	45
<i>Figure 3.24.</i> Auto-corrélation de l'erreur de prédiction.....	49
<i>Figure 3.25.</i> Evolution des paramètres (N =1024, $\delta^2 = 0.0001$).....	49
<i>Figure 3.26.</i> Evolution des paramètres (N=1024, $\delta^2=0.0001$)	50
<i>Figure 3.27.</i> Auto-corrélation de l'erreur de prédiction.....	50
<i>Figure 3.28.</i> Auto-corrélation de l'erreur de prédicti	52
<i>Figure 4.1.</i> Représentation de l'appareil phonatoire humain.....	56
<i>Figure 4.2.</i> Le signal de parole	56
<i>Figure 4.3.</i> Une partie voisée du signal de parole.....	59
<i>Figure 4.4.</i> Evolution des paramètres du signal de parole.....	60
<i>Figure 4.5.</i> Le signal de l'erreur de prédiction.....	60
<i>Figure 4.6.</i> Auto corrélation de l'erreur de prédiction.	61
<i>Figure 4.7.</i> Une partie non voisée du signal de parole	61
<i>Figure 4.8.</i> Evolution des paramètres du signal de parole.....	62
<i>Figure 4.9.</i> Signal de l'erreur de prédiction.....	63
<i>Figure 4.10.</i> Auto corrélation de l'erreur de prédiction	63

Liste des tableaux

<i>Tableau 3.1.</i>	Influence du nombre d'échantillons pour le 1 ^{er} système.	30
<i>Tableau 3.2.</i>	Influence de nombre d'échantillons N pour le 2 ^{ème} système.	31
<i>Tableau 3.3.</i>	Influence du nombre d'échantillons N pour le 3 ^{ème} système.....	32
<i>Tableau 3.4.</i>	Influence du nombre d'échantillons N pour le 4 ^{ème} système.....	34
<i>Tableau 3.5.</i>	Influence de la variance δ^2 pour le 1 ^{er} système.	35
<i>Tableau 3.6.</i>	Influence de la variance δ^2 pour le 2 ^{ème} système.....	36
<i>Tableau 3.7.</i>	Influence de la variance δ^2 pour le 3 ^{ème} système.....	36
<i>Tableau 3.8.</i>	Influence de la variance δ^2 pour le 4 ^{ème} système.	36
<i>Tableau 3.9.</i>	Influence du facteur d'oubli pour le 1 ^{er} système.....	37
<i>Tableau 3.10.</i>	Influence de facteur d'oubli pour le 2 ^{ème} systeme.	38
<i>Tableau3.11.</i>	Influence du facteur d'oubli pour le 3 ^{ème} système	39
<i>Tableau 3.12.</i>	Influence de facteur d'oubli pour 4 ^{ème} système.	40
<i>Tableau 3.13.</i>	Influence du nombre d'échantillons 1 ^{er} système avec $\delta^2 = 0.001$	42
<i>Tableau 3.14.</i>	Influence du nombre d'échantillons 2 ^{ème} système avec $\delta^2 = 0.001$	43
<i>Tableau 3.15.</i>	Influence du nombre d'échantillons 3 ^{ème} système avec $\delta^2 = 0.001$	44
<i>Tableau 3.16.</i>	Influence du nombre d'échantillons pour le 4 ^{ème} système avec $\delta^2 = 0.001$	45
<i>Tableau 3.17.</i>	Influence de la variance pour le 1 ^{er} système avec $N=256*4$	46
<i>Tableau 3.18.</i>	Influence de la variance pour le 2 ^{ème} système avec $N=256*4$	46
<i>Tableau 3.19.</i>	Influence de la variance pour le 3 ^{ème} système avec $N=256*4$	47
<i>Tableau 3.20.</i>	Influence de la variance pour le 4 ^{ème} système avec $N=256*4$	47
<i>Tableau 3.21.</i>	Influence du nombre d'échantillons pour le 1 ^{er} système avec $\delta^2 = 0.0001$	48
<i>Tableau 3.22.</i>	Influence du nombre d'échantillons pour le 2 ^{ème} système avec $\delta^2 = 0.0001$. ..	49
<i>Tableau 3.23.</i>	Influence de la variance pour le 1 ^{ere} système avec $N=256*4$	50
<i>Tableau 3.24.</i>	Influence de la variance pour le 2 ^{ème} système avec $N=256*4$	51
<i>Tableau 3.25.</i>	Influence du nombre d'échantillons pour le 1 ^{er} système avec $\delta^2 = 0.0001$	52
<i>Tableau 3.26.</i>	Influence du nombre d'échantillons pour le 2 ^{ème} système avec $\delta^2 = 0.0001$. ..	53
<i>Tableau 3.27.</i>	Influence de la variance pour le 1 ^{er} système avec $N=256*4$	53
<i>Tableau 3.28.</i>	Influence de la variance pour le 2 ^{ème} système avec $N=256*4$	53

<i>Tableau 4.1.</i> Estimation des paramètres du signal de parole.....	59
<i>Tableau 4.2.</i> Estimation des paramètres du signal de parole.	62
Tableau 4.3. Estimation des paramètres d'un signal de parole en présence du bruit. (Partie non voisé)	64

Introduction générale

L'intérêt des approches d'identification paramétrique fait l'objet de nombreux développements dans ces dernières années. Les résultats théoriques établis concernent généralement les performances asymptotiques des méthodes dans le cas de signaux d'entrée connus et suffisamment excitants. Le cas plus réaliste de données faiblement informatives, en nombre fini ou avec des incertitudes non seulement sur les signaux de sortie mais aussi sur les signaux d'entrée a reçu moins d'attention jusqu'à présent ; de ce fait, l'objectif de l'identification est alors de fournir une estimation des paramètres du modèle mathématique du système considéré afin de le simuler, de le commander ou de détecter un défaut de fonctionnement. En effet, l'identification consiste à identifier les paramètres d'un système pour voir l'évolution des grandeurs de sorties suivant les variations d'entrées. Pour ce faire il est intéressant de synthétiser un modèle mathématique dont le comportement soit identique à celui du système étudié.

Dans ce mémoire de projet fin d'étude, nous nous sommes intéressés à l'étude et implémentation sous MATLAB, la technique d'identification paramétrique basée sur la prédiction, la méthode utilisée est la méthode des moindres carrés récursive appliquée aux modèle stochastique ARX, ARMAX, BJ et OE , basés sur les données d'observations. En suite, nous allons tester les performances de la méthode proposée, et de faire plusieurs tests de simulation en langage MATLAB.

Le premier chapitre, est réservé pour la modélisation et l'identification des systèmes ainsi que les différents types de modèle stochastiques (ARX, ARMAX, BJ, OE).

Le deuxième chapitre, nous allons faire une étude détaillée sur les algorithmes de la méthode de moindre carré (RLS, λ RLS, RELS, RLS appliquée au modèle BJ et RLS appliquée au modèle OE).

Le troisième chapitre est consacré à l'implémentation des algorithmes d'identification et aux résultats obtenus par simulation. Plusieurs tests ont été effectués sur différents processus physiques stables à phase minimale ou non minimale ainsi que leurs interprétations physiques.

Dans le quatrième chapitre, nous présenterons quelques généralités sur le traitement automatique de la parole afin de faire une application de l'algorithme des moindres carrés récursive (modèle AR) sur un signal de parole.

Enfin de ce mémoire, une conclusion générale résumant les différentes méthodes d'identification développées dans ce travail et nous y mettons l'accent aussi sur les futures recherches s'inscrivant dans le prolongement de ce travail et à mener ultérieurement.

Chapitre 1 Modélisation et Identification

1.1 Introduction

Dans ce chapitre, nous nous sommes intéressés à la modélisation et l'identification des systèmes. La modélisation consiste à trouver la relation entre la sortie et l'entrée de système, par contre l'identification c'est l'opération de détermination des caractéristiques dynamiques d'un système, c'est-à-dire la détermination des paramètres inconnus du modèle pour avoir un modèle très proche au système.

1.2 Modélisation

1.2.1 Définition

La modélisation est la représentation d'un système par un modèle plus facile à appréhender. Il peut s'agir d'un modèle de connaissance ou de représentation. Le modèle sera alors numérique ou analogique. La modélisation d'un processus consiste à trouver un modèle dont le comportement dynamique se rapproche à celui du processus. Ce modèle sera utilisé pour effectuer des prédictions de la sortie du processus, ou pour l'apprentissage d'un correcteur, ou encore pour simuler le processus au sein d'un système de commande [6].

1.2.2 But de la modélisation

La modélisation a pour but de chercher une représentation mathématique susceptible de se comporter comme le système physique, si modèle et le processus sont soumis à la même sollicitation d'entrée [7].

1.2.3 Modèles mathématiques

a) Définition

Un modèle d'un système traduit les relations entre les différentes variables de ce système par des équations, nous nous intéressons ici aux modèles mathématiques, qui représentent la description de système en termes d'expressions mathématiques comme des équations différentielles (systèmes continus) ou des équations aux différences (systèmes discrets). Le type d'expressions utilisées donne des qualificatifs au modèle: linéaire / non-linéaire, continu / discret, déterministe / stochastique obtenu par une formulation mathématique entre les différentes variables du système.

b) Intérêts

Un modèle est utilisé soit pour :

- La simulation; Exemple simulateur de vol.
- La prédiction des sorties; Exemple : prévision d'évolution des actions dans le marché, prévision météorologique, commande prédictive, etc.
- La synthèse des régulateurs; Exemple: contrôle de procédés industriels.

c) Conception

Un modèle mathématique est obtenu via deux approches :

- Modèle de connaissances ou modèle boîte grise (Grey-box). Les modèles de connaissances sont construits à partir d'une analyse physique, chimique, etc., ces modèles ne comportent généralement pas de paramètres ajustables, ou paramètres ajustables en très petit nombre. Les modèles de connaissances sont familiers à tous ceux qui ont suivi des cours de physique ou de chimie, donc les paramètres de modèle de connaissances ont un sens physique [8], (résistance électrique, longueur,...).
- Modèle de comportement entrées-sorties ou modèle boîte noire (Black-box). Les modèles n'ont aucun pouvoir explicatif de la structure physique de l'objet, leur structure n'est qu'une relation mathématique qui va relier localement les mesures des différentes variables du processus. Ces modèles de représentation

sont de type boîte noire, et les paramètres donc n'ont aucun sens physique mais sont suffisants dans les problèmes de traitement de signal. Les modèles de représentation sont en générale assez simples à simuler et mieux adaptés à la détermination d'une commande, ils sont d'utilisations très fréquentes [9].

1.2.4 Principaux étapes de la modélisation

On distingue six étapes essentielles de la modélisation qui sont :

a) *Systeme*

Un système (processus) sera pour nous une partie de l'univers qui nous entoure, et qui nous avons décidé, plus ou moins arbitrairement d'appréhender comme un tout avec lequel nous interagissons. Nous observons certaines grandeurs caractéristiques du système et le résultat de ces observations forme le vecteur des sorties y peuvent être connues et maîtrisable que nous noterons u , ou non maîtrisable et plus ou moins inconnus, ce sont les perturbations ou bruit que nous noterons p , (voir Figure 1.1).



Figure 1.1. Représentation d'un système.

b) *Modèle*

Le modèle d'un système physique est une règle permettant de calculer, à partir de grandeurs connues ou mesurées, d'autres grandeurs dont nous espérons qu'elles ressembleront aux grandeurs du système qui nous intéresse. Fréquemment, le modèle calcul, à partir de l'entrée u de système une sortie \hat{y} qui ressemble le plus possible à y . Si le modèle et le système ont alors la même entrée, on parlera de modèle parallèle (Figure 1.2), par contre dans le cas où le modèle calcule à partir de la sortie y un vecteur \hat{u} dont on souhaite qu'il ressemble le plus possible aux entrées u du système, on parlera du modèle série ou inverse (Figure 1.3).



Figure 1.2. Modèle parallèle.



Figure 1.3. Modèle série.

c) Critère

Pour fixer les idées, nous supposons que le modèle de type parallèle, c'est-à-dire, soumis aux mêmes entrées et aux mêmes conditions initiales que le système, on appelle alors erreur de sortie, la différence entre la sortie du système et celle du modèle (voir Figure 1.4). L'erreur de sortie peut s'écrire sous la forme :

$$err = \hat{y}(t) - y(t)$$

Le plus souvent, on souhaite que cette erreur de sortie soit aussi proche que possible de zéros, l'échelle de valeur qui sera utilisé pour effectuer la comparaison prendra la forme d'une fonction scalaire $J(\theta)$ appelée « critère », nous supposons en général que ce critère est optimal si $J[M_1(\theta_1)] < J[M_2(\theta_2)]$. Avec θ vecteur des paramètres.

Le choix du critère doit traduire le but fixé par la modélisation quelque soit le critère choisi, il convient ensuite de l'optimiser.

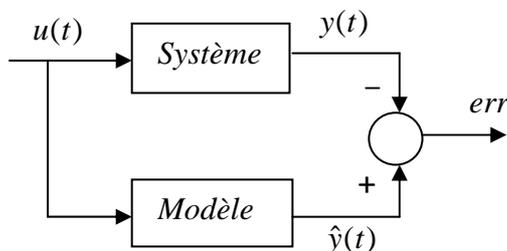


Figure 1.4. Erreur de sortie pour un modèle parallèle.

d) Optimiseur

L'algorithme d'optimisation reçoit toutes les informations disponibles et les utilise pour minimiser $J(\theta)$ de façon à calculer $\hat{\theta}$. Le schéma de circulation de l'information peut être celui décrit sur la Figure 1.5.

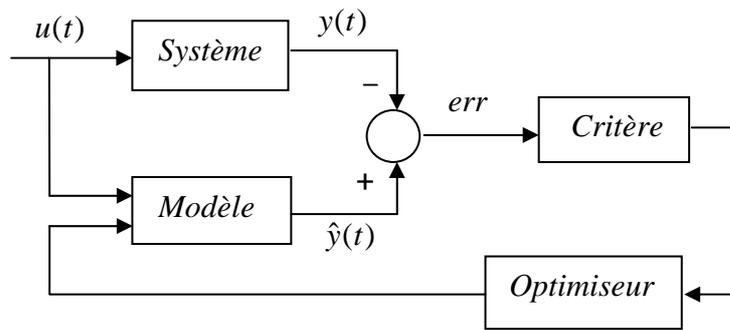


Figure 1.5. Exemple de circulation de l'information pour l'optimisation.

e) Incertitude sur les paramètres

La valeur de $\hat{\theta}$ obtenue avec l'algorithme d'optimisation correspond au meilleur modèle possible choisi parmi un ensemble de modèles acceptables qui dépend de l'expérience pour le recueil des données, donc on pourra faire la planification d'expérience, de façon à essayer de recueillir l'information la plus pertinente possible au sens du critère utilisé.

f) Analyse critique des résultats obtenus

La phase d'analyse critique des résultats obtenus est indispensable, il faut donc soumettre le modèle à un ensemble d'épreuves destinées à le valider, si l'on parvient ainsi à détecter des erreurs, alors on a un mauvais choix du modèle.

1.2.5 Modélisation autorégressive à moyenne ajustée (ARMA)

Un modèle autorégressif à moyenne ajustée d'ordre (n, m) , noté ARMA (n, m) est définie par l'équation aux différences suivante [16] :

$$\sum_{i=0}^n a_i y(t-i) = \sum_{i=0}^m b_i u(t-i) + e(t) \quad (1.1)$$

avec $a_0 = 1$, $b_0 = 0$

$u(t)$: Entrée du système

$y(t)$: Sortie du système

$e(t)$: Bruit blanc.

Donc par simple transformée en Z du modèle ARMA, on obtient:

$$y(t) = \frac{B(z)}{A(z)}u(t) + \frac{1}{A(z)}e(t) \quad (1.2)$$

avec

$H(z)$: La fonction de transfert du modèle.

$A(z)$: La transformée en Z de la partie AR.

$B(z)$: La transformée en Z de la partie MA.

1.2.6 Modèle ARMAX

Le modèle ARMAX est décrit par l'équation aux différences suivante :

$$\sum_{i=0}^n a_i y(t-i) = \sum_{i=0}^m b_i u(t-i) + v(t) \quad (1.3)$$

Où $v(t) = e(t) + c_1 e(t-1) + \dots + c_p e(t-p)$, est un bruit blanc filtré ($c_0 = 1$).

En utilisant la transformée en Z, l'équation (1.3) peut s'écrire :

$$\sum_{i=0}^n a_i z^{-i} Y(z) = \sum_{i=0}^m b_i z^{-i} U(z) + \sum_{i=0}^p c_i z^{-i} E(z) \quad (1.4)$$

Donc,

$$y(t) = \frac{B(z)}{A(z)}u(t) + \frac{C(z)}{A(z)}e(t) \quad (1.5)$$

Nous posons :

$$H_1(z) = \frac{B(z)}{A(z)} ; H_2(z) = \frac{C(z)}{A(z)}$$

avec

$H_1(z)$: Fonction de transfert du système.

$H_2(z)$: Fonction de transfert du filtre.

1.2.7 Modèle à erreur de sortie (OE)

Ce modèle suppose l'existence d'un bruit blanc agissant directement sur la sortie du système. Le modèle OE est décrit par l'équation suivante :

$$\sum_{i=0}^n a_i y(t-i) = \sum_{i=0}^m b_i u(t-i) + \sum_{i=0}^n a_i e(t-i) \quad (1.6)$$

En utilisant la transformée en Z, l'équation peut s'écrire :

$$\sum_{i=0}^n a_i z^{-i} Y(z) = \sum_{i=0}^m b_i z^{-i} U(z) + \sum_{i=0}^n a_i z^{-i} E(z) \quad (1.7)$$

Donc,

$$y(t) = \frac{B(z)}{A(z)} u(t) + e(t) \quad (1.8)$$

1.2.8 Modèle Box-Jenkins (bj)

Ce modèle suppose que la dynamique du bruit peut être différente à celle du système. Le modèle bj est décrit donc par l'équation aux différences suivante :

$$y(t) = \frac{B(z)}{A(z)} u(t) + \frac{C(z)}{D(z)} e(t) \quad (1.8)$$

Avec

$$A(z) = \sum_{i=0}^n a_i z^{-i}$$

$$B(z) = \sum_{i=0}^m b_i z^{-i}$$

$$C(z) = \sum_{i=0}^p c_i z^{-i}$$

$$D(z) = \sum_{i=0}^q d_i z^{-i}$$

1.3 Identification

1.3.1 Définition

Identifier un système dynamique réel c'est caractériser un autre système (appelé modèle), à partir de la connaissance expérimentale des entrées et sorties de manière à obtenir le même comportement. Donc, l'identification c'est l'opération de détermination des caractéristiques dynamiques d'un procédé (système). Dans cette partie, nous nous sommes intéressés à l'identification des modèles dynamiques paramétriques échantillonnés qui sont les plus appropriés pour la conception et l'ajustement des systèmes numériques de commande. On distingue deux types de l'identification :

- Identification non paramétrique (classique)
- Identification paramétrique [5]

a) Identification non paramétrique

Cette méthode initialement est utilisée pour obtenir des modèles paramétriques à partir des méthodes non paramétriques de type réponse à échelon [4].

b) Identification paramétrique

Cette approche moderne élimine tous les défauts des méthodes classiques et offre d'autres possibilités telles que [4] :

- Modélisation des bruits.
- Identification des modèles de perturbation.
- Détection et mesure des fréquences d'oscillations
- Analyse spectrales des signaux.

1.3.2 Les étapes d'identification

Pour parvenir à un bon modèle, nous devons généralement suivre les quatre étapes suivantes [5] :

a) Acquisition des entrées-sorties sous un protocole d'expérimentation

Il s'agit essentiellement de choisir un signal d'excitation avec une densité spectrale homogène couvrant l'ensemble de la bande passante du procédé à identifier. En pratique, nous utilisons deux catégories de signaux de tests :

- Les signaux déterministes tel que l'échelon, la sinusoïde etc. ces signaux sont décrit par une fonction de temps.
- Les signaux aléatoires, complètement décrit par leurs propriétés statistiques une des signaux les plus utilisés pour l'identification est la séquence pseudo aléatoire (SBPA).

En résumé l'entrée du système n'est intéressante que si elle est centrée, riche en fréquence et déterministe si c'est possible.

b) Choix ou estimation de la complexité du modèle

Le problème typique rencontré dans le cas du modèle paramétrique est le choix de l'ordre des polynômes (numérateur et dénominateur) de la fonction de transfert, ce choix de la complexité peut se faire par une procédure essais/erreur. On peut maintenant disposer des algorithmes qui estiment à partir des données, la complexité des modèles.

c) Estimation des paramètres du modèle

Une fois la complexité du modèle est fixée, donc il s'agit d'estimer les paramètres du modèle de façon à minimiser un critère représentatif. La qualité de cette estimation dépendra donc de la méthode choisie, et de l'information contenue dans les données d'entrée-sortie.

d) Validation du modèle

Cette étape est certainement la plus importante lors d'une identification, elle consiste à accepter ou rejeter le modèle obtenu, le rejet d'un modèle qui ne répond pas à ses objectifs remet en cause l'ensemble des étapes précédentes. Ainsi pour l'analyse de la validité du modèle, on teste par exemple, le blanchissement de l'erreur d'identification par le calcul de sa fonction d'auto-corrélation.

1.3.3 Algorithme d'identification récursif

Dans les algorithmes récursifs, les paramètres estimés sont optimisés progressivement en utilisant chaque fois une seule paire de données entrée/sortie. Les algorithmes récursifs présentent un certain nombre d'avantages [5] :

- Une estimation récursive du modèle peut être obtenue au fur et à mesure que le système évolue, elle est donc adaptée à l'identification en temps réel.
- Utilisation d'une taille mémoire réduite.
- Le modèle estimé en ligne contient toutes les informations contenues dans les données entrée /sorties passées.
- Ils peuvent être aussi utilisés pour l'identification non récursive

1.4 Conclusion

Dans ce chapitre, nous avons présenté quelques notions très importantes sur la modélisation et l'identification, l'efficacité des modèles repose sur analogique, entre le comportement des objets physiques et celui des êtres mathématiques. La modélisation permet de représenter sous forme synthétique et cohérente un ensemble de connaissances ; l'utilisation de matériel informatique (ordinateur ou microprocesseur) impose de travailler avec des valeurs discrètes des différents signaux. L'identification consiste à ajuster le paramètre inconnu du modèle de manière à ce que celui-ci décrive au mieux le fonctionnement du procédé, elle comporte quatre étapes :

- Acquisition des entrées/sorties sous un protocole d'expérimentation.
- Choix de structure du modèle.
- Estimation des paramètres du modèle.
- Validation du modèle identifié

Dans le chapitre qui suit, nous aborderons les méthodes d'identification de modèles paramétriques qui décrivent l'évolution des procédés au cours du temps, ainsi que la présentation des différentes structures de modèles vis-à-vis du bruit de sortie du système tel que (ARX, ARMAX, BJ, OE). Ensuite nous allons étudier en détails

les algorithmes de moindres carrés (RLS, λ RLS, RELS,...) qui sont la base de toutes les méthodes de l'identification et de l'estimation des paramètres.

Chapitre 2 L'algorithme des moindres carrés

2.1 Introduction

Dans ce chapitre on va étudier les algorithmes d'identification pour identifier les paramètres d'un système physique. Ces algorithmes ont pour fonction d'ajuster un vecteur de paramètres, que nous noterons généralement θ . Il est important de souligner qu'il n'y a pas un Algorithme d'estimation paramétrique unique pour tous les types de modèles de bruit fournit des estimations paramétriques asymptotiquement non biaisées. Pour chaque structure de bruit, il existe des algorithmes spécifiques permettant d'obtenir des bons résultats, Il convient tout d'abord de préciser les principaux modèles « Système + perturbation » [4].

Avant de présenter l'algorithme des moindres carrés étendus RELS, nous présenterons une étude détaillée de l'algorithme des moindres carrés récursif RLS et l'algorithme des moindres carrés pondéré λ RLS et l'algorithme des moindres carrés basé sur les modèles BJ et OE.

2.2 Définition de la méthode des moindres carrés

La méthode des moindres carrés est introduite par Karl Gauss en 1809 dans le but d'extraire des paramètres de mesures entachées de bruit, La méthode des moindres carrés permet alors de minimiser l'impact des erreurs expérimentales en « ajoutant de l'information » dans le processus de mesure, cette technique est aujourd'hui largement utilisée dans de nombreux domaines, Elle est basée sur la minimisation d'une fonction quadratique J correspond au carré de l'instant courant, entre la sortie du modèle et la valeur de la sortie du processus ou celle du signal que l'on veut modéliser la fonction J et définie comme suit [11] :

$$J_N() = \frac{1}{N} \sum_{t=1}^N [e(t)]^2 \quad (2.1)$$

Avec $e(t)$ représente l'erreur de prédiction et N le nombre d'échantillons.

2.2.1 Identification du modèle ARMA

La modélisation autorégressive à moyenne ajustée d'ordre (n,m) noté ARMA (n,m) est définie par l'équation aux différences suivante :

$$\sum_{i=0}^n a_i y(t-i) = \sum_{i=0}^m b_i u(t-i) + e(t) \quad (2.2)$$

Avec $a_0 = 1, b_0 = 0$

$u(t)$: Entrée du système

$y(t)$: Sortie du système

$e(t)$: Séquence de variables aléatoires indépendantes de moyenne nulle et de variance δ^2 en utilisant la transformée Z, L'équation (2.2) peut s'écrire :

$$\sum_{i=0}^n a_i z^{-i} Y(z) = \sum_{i=0}^m b_i z^{-i} U(z) + E(z) \quad (2.3)$$

On pose: $A(Z) = \sum_{i=0}^n a_i z^{-i}$ et $B(Z) = \sum_{i=0}^m b_i z^{-i}$

Donc, l'équation (2.3) s'écrire sous cette forme:

$$A(Z)Y(Z) = B(Z)U(Z) + E(Z) \quad (2.4)$$

Ce qui donne comme erreur :

$$E(Z) = A(Z)Y(Z) - B(Z)U(Z) \quad (2.5)$$

A partir des deux équations (2.4) et (2.5), on peut tirer le schéma suivant :

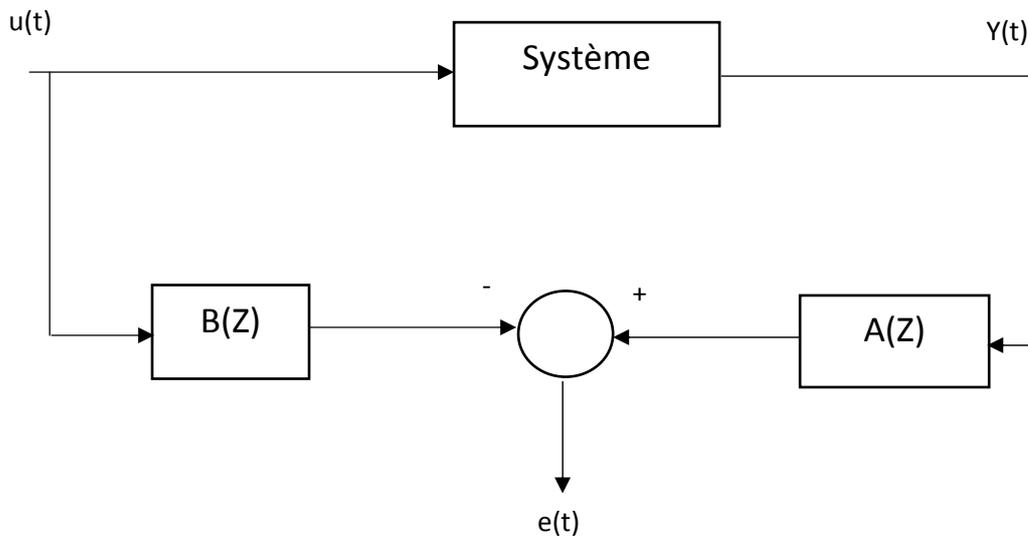


Figure 2.1. Structure de réalisation d'un modèle ARMA.

2.2.2 Solution optimale au sens des moindres carrés

On peut donc, penser de manière intuitive que si l'on augmente le nombre d'observation, le problème se ramènera à la résolution d'un système d'équation linéaire. Dans le cas d'un système de type ARMA, nous effectuons N mesures d'observation [11], nous pouvons écrire d'après l'équation (2.2), l'équation suivante :

$$t = 1 : y(1) = -a_1y(0) - a_2y(-1) - \dots - a_ny(1-n) + b_1u(0) + \dots + b_mu(1-m) + e(1)$$

$$t = 2 : y(2) = -a_1y(1) - a_2y(0) - \dots - a_ny(2-n) + b_1u(1) + \dots + b_mu(2-m) + e(2)$$

$$t = N : y(N) = -a_1y(N-1) - \dots - a_ny(N-n) + b_1u(N-1) + \dots + b_mu(N-m) + e(N)$$

La représentation matricielle d'un système peut s'écrire sous forme d'une équation de modélisation [11] :

$$y(t) = T(t) + e(t) \tag{2.6}$$

Avec $\mathbb{T}(t)$ est appelée le vecteur de données, θ est le vecteur des paramètres à identifier et $e(t)$ un bruit blanc.

$$T = [a_1 \ a_2 \ \dots \ a_n, b_1 \ b_2 \ \dots \ b_m]$$

$$T(t) = [-y(t-1) \ \dots -y(t-n), u(t-1) \ \dots \ u(t-m)]$$

On définit l'erreur de prédiction comme étant la différence entre la sortie du système et la sortie du modèle:

$$e(t) = y(t) - \hat{y}(t) \quad (2.7)$$

Avec
$$\hat{y}(t) = \hat{\gamma}^T(t)\phi(t-1) \quad (2.8)$$

Avec $\hat{\gamma}(t-1)$: Vecteur des paramètres estimés.

La minimisation du critère $J_N(\theta)$ consiste à trouver un optimum, c'est-à-dire de calculer sa dérivée :

$$\left[\frac{\partial J_N(\theta)}{\partial \theta} \right]_{\theta=\hat{\gamma}(N)} = 0 \quad (2.9)$$

$$\left[\frac{\partial J_N(\theta)}{\partial \theta} \right]_{\theta=\hat{\gamma}(N)} = -\frac{2}{N} \sum_{t=1}^N \phi(t) [y(t) - \hat{\gamma}^T(t)\phi(t)] \quad (2.10)$$

A partir de ces deux dernières équations (2.9) et (2.10), on déduit la solution optimale au sens des moindres carrés de la forme suivante :

$$\hat{\gamma}(N) = \left[\sum_{t=1}^N \phi(t)^T \phi(t) \right]^{-1} \sum_{t=1}^N \phi(t) y(t) \quad (2.11)$$

Nous constatons que la matrice $\left[\sum_{t=1}^N \phi(t)^T \phi(t) \right]$ est grande, si le nombre d'échantillons N est important, d'où le calcul de son inverse n'est pas conseillé, pour cela on utilise l'estimation des moindres carrés.

2.2.3 Algorithme des moindres carrés récurrents (RLS)

Pour la mise en œuvre de l'algorithme des moindres carrés récurrents, [11] :

On pose : $R(t) = \sum_{k=1}^t \phi(k)^T \phi(k)$

$$R(t) = R(t-1) + \phi(t)^T \phi(t) \quad (2.12)$$

D'après l'équation (2.11) et (2.12), on a :

$$\hat{\gamma}(t) = R^{-1}(t) \sum_{k=1}^t \phi(k) y(k) \quad (2.13)$$

$$\hat{\gamma}(t) = R^{-1}(t) \left[\sum_{k=1}^{t-1} \phi(k) y(k) + \phi(t) y(t) \right] \quad (2.14)$$

$$\hat{\gamma}(t) = R^{-1}(t) [R(t-1) \hat{\gamma}(t-1) + \phi(t) y(t)] \quad (2.15)$$

$$\hat{\gamma}(t) = R^{-1}(t) [R(t) \hat{\gamma}(t-1) - \phi(t)^T \phi(t) \hat{\gamma}(t-1) + \phi(t) y(t)] \quad (2.16)$$

$$\hat{\gamma}(t) = \hat{\gamma}(t-1) + R^{-1}(t) \phi(t) [y(t) - \phi(t)^T \hat{\gamma}(t-1)] \quad (2.17)$$

D'après cette dernière équation (2.17), on remarque que la solution des moindres carrés récursive contient le terme $R^{-1}(t)$ qui nécessite une inversion matricielle à chaque instant t . Rappelons le lemme d'inversion matricielle qui se présente sous la forme :

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B + C^{-1}]^{-1}D \quad (2.18)$$

Nous posons:

$$\begin{aligned} A &= R(t-1), B = \phi(t), C = 1, D = \phi(t)^T \\ R^{-1}(t) &= [R(t-1) + \phi(t)^T \phi(t)]^{-1} \end{aligned} \quad (2.19)$$

En utilisant le lemme d'inversion matricielle, l'équation (2.12) peut se réécrire :

$$R^{-1}(t) = R^{-1}(t-1) - \frac{R^{-1}(t-1) \phi(t)^T \phi(t) R^{-1}(t-1)}{(1 + \phi(t)^T R^{-1}(t-1) \phi(t))} \quad (2.20)$$

L'introduction de la matrice du gain d'adaptation $p(t) = R^{-1}(t)$, permet la mise en œuvre de l'algorithme des moindres carrés RLS de la forme suivante :

$$\begin{cases} \hat{\gamma}(t) = \hat{\gamma}(t-1) + P(t) \phi(t) (y(t) - \phi(t)^T \hat{\gamma}(t-1)) \end{cases} \quad (2.21)$$

$$\begin{cases} P(t) = p(t-1) - \frac{p(t-1) \phi(t)^T \phi(t) p(t-1)}{1 + \phi(t)^T p(t-1) \phi(t)} \end{cases} \quad (2.22)$$

2.2.4 Algorithme des moindres carrés récursif pondéré λ RLS

Il existe d'autres versions de l'algorithme RLS, particulièrement une version permettant d'intégrer un facteur d'oubli exponentiel λ pour ne plus prendre en compte les premières itérations qui sont souvent peu significatives. L'idée de base qui permet cette adaptation consiste à observer les données à travers une fenêtre de pondération. Le rôle de

cette fenêtre est de pondérer les observations en fonction de la crédibilité qu'on leur attribue [13].

Si $\lambda=1$, on retrouve l'algorithme classique RLS, et dans le cas contraire, les erreurs sont pondérées par un coefficient λ^{t-k} . D'où l'équation (2.12) peut se réécrire :

$$R(t) = \sum_{k=1}^t \lambda^{t-k} (k)^T (k) = \lambda R(t-1) + (t)^T (t) \quad (2.23)$$

Nous rappelons l'équation (2.17) :

$$\hat{\gamma}(t) = \hat{\gamma}(t-1) + R^{-1}(t) (t) [y(t) - \mathcal{X}^T(t-1) (t)] \quad (2.24)$$

En utilisant le lemme d'inversion matricielle, l'équation (2.23) peut se réécrire :

$$R^{-1}(t) = \lambda^{-1} \left[R^{-1}(t-1) - \frac{R^{-1}(t-1) (t)^T (t) R^{-1}(t-1)}{\lambda + \mathcal{X}^T(t) R^{-1}(t-1) (t)} \right] \quad (2.25)$$

L'introduction de la matrice du gain d'adaptation $P(t) = R^{-1}(t)$, permet la mise en œuvre de l'algorithme des moindres carrés récursif pondéré λ RLS de la forme suivante :

$$\begin{cases} \hat{\gamma}(t) = \hat{\gamma}(t-1) + P(t) (t) (y(t) - \mathcal{X}^T(t-1) (t)) \end{cases} \quad (2.26)$$

$$\begin{cases} P(t) = \lambda^{-1} \left[p(t-1) - \frac{p(t-1) (t)^T (t) p(t-1)}{\lambda + \mathcal{X}^T(t) p(t-1) (t)} \right] \end{cases} \quad (2.27)$$

2.2.5 Propriétés de l'estimateur des moindres carrés

a Définition du biais

En l'absence de bruit, moyennant la condition d'excitation persistante, l'estimateur des moindres carrés tend vers la vraie valeur de γ lorsque $N \rightarrow +\infty$. En présence de bruit nous avons une erreur d'estimation qui définit le biais de l'estimateur : [14]

$$\text{Biais} = \lim_{N \rightarrow \infty} (\hat{\gamma}_N - \gamma) \quad (2.28)$$

Le biais peut se mettre sous la forme suivante :

$$\text{Biais} = \lim_{n \rightarrow +\infty} \left\{ - \left[\frac{1}{N} \sum_{t=1}^N (t)^T (t) \right]^{-1} \left[\frac{1}{N} \sum_{t=1}^N (t) e(t) \right] \right\} \quad (2.29)$$

Si nous introduisons l'espérance mathématique E , nous obtenons l'expression stochastique

$$\text{du biais : } \quad \text{Biais} = -E \left((t)^T (t) \right)^{-1} E((t)e(t)) \quad (2.30)$$

b Absence de biais

Pour que le biais soit nul, on pourrait avoir $E \left((t)^T (t) \right)^{-1} = 0$ ou $E((t)e(t)) = 0$ mais $(t) \neq 0$ donc il reste $E((t)e(t)) = 0$

Cette condition se traduit par [8]:

- $E((t)e(t)) = E \left((t) \right) E(e(t))$, donc (t) et $e(t)$ sont statistiquement indépendants (in-corrélé).
- $E(e(t)) = 0$ donc est un bruit centré.

En remplaçant l'équation (2.6) et (2.11), on obtient :

$$\hat{\gamma}(N) = \left[\sum_{t=1}^N (t)^T (t) \right]^{-1} \sum_{t=1}^N (t) [{}^T(t)_0 + e(t)] \quad (2.31)$$

$$\hat{\gamma}(N) = \gamma_0 + \left(\left[\sum_{t=1}^N (t)^T (t) \right]^{-1} \sum_{t=1}^N (t) e(t) \right) \quad (2.32)$$

2.3 Etude et développement de la méthode des moindres carrés étendus

2.3.1 Introduction

Cette méthode a été développée pour pouvoir identifier sans biais des modèles (procédé + perturbation) du type ARMAX dont la structure suivante [8]:

$$A(Z)Y(Z) = B(Z)U(Z) + C(Z)E(Z) \quad (2.33)$$

Rappelons que l'algorithme RLS suppose que le bruit additif en sortie soit un bruit blanc par contre dans le cas du RLSE considère que le bruit ajouté en sortie est un bruit coloré. Le bruit peut être considéré comme étant le résultat du filtrage d'un bruit blanc. L'idée est d'identifier simultanément le modèle du procédé et le modèle de la perturbation, pour pouvoir obtenir une erreur de prédiction asymptotiquement "blanche".

2.3.2 Algorithme des moindres carrés étendus

Le modèle ARMAX ayant la structure suivante :

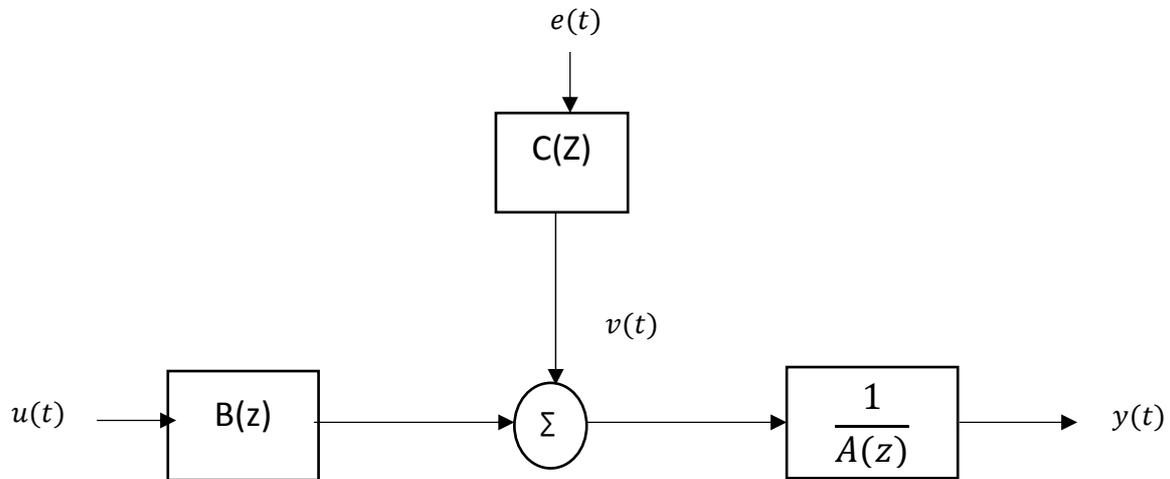


Figure 2.2. Structure de réalisation d'un modèle ARMAX [8].

Reprenons le modèle ARMAX décrit par l'équation suivante :

$$y(t) = \sum_{i=0}^m b_i u(t-i) - \sum_{i=1}^n a_i y(t-i) + v(t) \quad (2.34)$$

avec

(n.m) : ordre du modèle

$u(t)$: Entrée du système

$y(t)$: Sortie du système

$v(t)$: Bruit blanc filtré

La perturbation $v(t)$ est considérée comme le résultat d'un bruit blanc filtré, et on écrit

$$v(t) = e(t) + C_1 e(t-1) + \dots + C_p e(t-p) \quad (2.35)$$

En utilisant la transformée en L'équation (2.34) peut s'écrire :

$$\sum_{i=0}^n a_i z^{-i} Y(Z) = \sum_{i=0}^m b_i z^{-i} U(Z) + \sum_{i=0}^p c_i z^{-i} E(Z) \quad (2.36)$$

Nous posons : [16]

$$A(Z) = \sum_{i=0}^n a_i z^{-i} \quad (2.37)$$

$$B(Z) = \sum_{i=0}^m b_i z^{-i} \quad (2.38)$$

$$C(Z) = \sum_{i=0}^p c_i z^{-i} \quad (2.39)$$

avec $a_0=1, b_0=0, c_0=1$. [16]

L'équation (2.35) s'écrira donc:

$$Y(z) = \frac{B(Z)}{A(Z)} U(Z) + \frac{C(Z)}{A(Z)} E(Z) \quad (2.40)$$

avec $H_1(Z) = \frac{B(Z)}{A(Z)}$ et $H_2(Z) = \frac{C(Z)}{A(Z)}$

$H_1(Z), H_2(Z)$ sont des fonctions de transfert du système et de filtre respectivement.

2.3.3 Détermination des paramètres

On peut donc, penser de manière intuitive que si l'on augmente le nombre d'observation, le problème se ramènera à résolution d'un système d'équation linéaire.

Dans le cas d'un système bruité, nous effectuons N mesures d'observations et nous pouvons écrire d'après l'équation (2.33), une forme matricielle :

$$Y(t) = T(t-1) (t) + e(t) \quad (2.41)$$

Posons pour la commodité des notations :

$$T = [a_1 \ a_2 \ \dots \ a_n, b_1 \ b_2 \ \dots \ b_m, c_1 \ \dots \ c_p]$$

$$T(t) = [-y(t-1) \ \dots \ -y(t-n), u(t-1) \ \dots \ u(t-m), e(t-1) \ \dots \ e(t-p)]$$

avec

θ^T : Vecteur des paramètres à identifier

(t) : Vecteur des données.

T : désigne la transpose d'un vecteur ou d'une matrice.

On définit l'erreur de prédiction a priori comme étant la différence entre la sortie du système et la sortie du modèle :

$$e(t) = y(t) - \hat{y}(t) \quad (2.42)$$

Sachant que :

$$\hat{y}(t) = \theta^T(t-1) \quad (2.43)$$

$\theta(t-1)$: Représente les paramètres estimés.

2.3.4 Equations de la mise en œuvre de l'algorithme RELS

Le problème de la méthode des moindres carrés ordinaire c'est que $e(t)$ n'est pas mesurable, donc il faut l'estimer. D'après l'équation (2.33) et (2.34) on écrit : [8]

$$e(t) = y(t) + a_1 y(t-1) + \dots + a_n y(t-n) - b_1 u(t-1) - \dots - b_m u(t-m) - c_1 e(t-1) - \dots - c_p e(t-p) \quad (2.44)$$

On appelle $\hat{y}(t)$ l'estimateur de $e(t)$, d'où l'équation (2.44) s'écrira sous la forme :

$$\hat{y}(t) = y(t) + \hat{a}_1 y(t-1) + \dots + \hat{a}_n y(t-n) - \hat{b}_1 u(t-1) - \dots - \hat{b}_m u(t-1) - \hat{c}_1 e(t-1) - \dots - \hat{c}_p e(t-p) \quad (2.45)$$

La prédiction, ajustable a priori dans le cas des moindres carrés étendus s'obtient de l'équation (2.42), en remplaçant les paramètres connus par les paramètres estimés :

$$\theta^T = [\hat{a}_1 \dots \hat{a}_n \quad \hat{b}_1 \dots \hat{b}_m \quad \hat{c}_1 \dots \hat{c}_p]$$

$$T(t) = [-y(t) \quad \dots \quad -y(t-n) \quad u(t) \quad \dots \quad u(t-m) \quad (t) \quad \dots \quad (t-p)]$$

En résumé les équations permettant la mise en œuvre de l'algorithme des moindres carrés étendus (RELS) seront :

$$\begin{cases} \hat{\gamma}(t) = y(t) - \hat{\gamma}^T(t-1) p(t-1) & (2.46) \\ \hat{\gamma}(t) = \hat{\gamma}(t-1) + P(t) p(t) (y(t) - \hat{\gamma}^T(t-1) p(t-1)) & (2.47) \\ P(t) = p(t-1) - \frac{p(t-1) p^T(t-1) p(t-1)}{1 + p^T(t-1) p(t-1)} & (2.48) \end{cases}$$

2.3.5 Initialisation de l'algorithme RELS

L'initialisation concernant la matrice $p(t)$ et le vecteur des paramètres $\hat{\gamma}(t)$ se fait souvent comme suite [8]:

$$\begin{cases} \hat{\theta}(0) = \theta_p \text{ à priori} \\ \hat{\theta}(0) = 0 \text{ sinon} \end{cases}$$

$P(0) = C.I$ tel que C : Constant et I : Matrice d'identité.

2.4 Algorithme des moindres carrés avec modèle OE

2.4.1 Le modèle avec erreur de sortie (OE)

$$y(t) = \frac{B(Z)}{A(Z)} u(t) + e(t) \quad (2.49)$$

Avec

$$A(Z) = \sum_{i=0}^n a_i z^{-i}; B(Z) = \sum_{i=0}^m b_i z^{-i}; B(Z) = \sum_{i=0}^m b_i z^{-i}$$

et $a_0 = 1, b_0 = 0$

2.4.2 Le prédicteur d'un modèle OE

Le prédicteur optimal pour ce modèle :

On a la relation entre la sortie de système et la sortie estimée :

$$y(t) = \hat{y}(t) + e(t) \quad (2.50)$$

Démonstration : par l'identification entre l'équation (2.49) et (2.50) on obtient

$$\hat{y}(t) = \frac{B(Z)}{A(Z)} u(t) \quad (2.51)$$

2.4.3 La forme matricielle

On peut écrire l'équation (2.49) par la forme matricielle suivant :

$$y(t) = T(t-1) \hat{\gamma}(t) + e(t) \quad (2.52)$$

Avec : $\hat{\gamma}(t) = T(t) \hat{\gamma}(t-1)$ est un sortie estimé

D'après l'équation (2.51) on a:

$$\hat{\gamma}(t) = \frac{B(Z)}{A(Z)} u(t) \rightarrow A(Z)\hat{\gamma}(t) = B(Z)u(t) \quad (2.53)$$

Donc

$$\hat{\gamma}(t) = - \sum_{i=1}^n a_i \hat{\gamma}(t-i) + \sum_{i=1}^m b_i u(t-i) \quad (2.54)$$

En résulté que :

$$T = [a_1 \ a_2 \ \dots \ a_n, b_1 \ \dots \ b_m]$$

$$T(t) = [-\hat{\gamma}(t-1) \ \dots \ -\hat{\gamma}(t-n), u(t) \ \dots \ u(t-m)]$$

d'après les équations de (2.7) à (2.20) on obtient :

$$\begin{cases} \hat{\gamma}(t) = \hat{\gamma}(t-1) + P(t) (y(t) - T(t-1) \hat{\gamma}(t-1)) \\ P(t) = p(t-1) - \frac{p(t-1) T(t-1)^T (y(t) - T(t-1) \hat{\gamma}(t-1))}{1 + T(t-1)^T p(t-1)} \end{cases} \quad (2.55)$$

$$\begin{cases} \hat{\gamma}(t) = \hat{\gamma}(t-1) + P(t) (y(t) - T(t-1) \hat{\gamma}(t-1)) \\ P(t) = p(t-1) - \frac{p(t-1) T(t-1)^T (y(t) - T(t-1) \hat{\gamma}(t-1))}{1 + T(t-1)^T p(t-1)} \end{cases} \quad (2.56)$$

2.5 Algorithme des moindres carrée avec modèle Box-Jenkins

2.5.1 Le modèle Box-Jenkins

$$y(t) = \frac{B(Z)}{A(Z)} u(t) + \frac{C(Z)}{D(Z)} e(t) \quad (2.57)$$

Avec

$$A(Z) = \sum_{i=0}^n a_i z^{-i} \quad \text{et} \quad B(Z) = \sum_{i=0}^m b_i z^{-i}$$

et

$$C(Z) = \sum_{i=0}^p c_i z^{-i} \quad \text{et} \quad D(Z) = \sum_{i=0}^q d_i z^{-i}$$

et $a_0 = 1, b_0 = 0, c_0 = 1, d_0 = 1$

Avec

$y(t)$: la sortie du système

$u(t)$: l'entrée du système.

$e(t)$: bruit blanc.

2.5.2 Le prédicteur d'un modèle BJ

Le prédicteur optimal pour ce modèle :

On a la relation entre la sortie de système et la sortie estimée :

$$y(t) = \hat{y}(t) + e(t) \quad (2.58)$$

On remplace l'équation (2.58) dans l'équation (2.57) et on obtient le prédicteur suivants :

$$\hat{y}(t) = \frac{D(Z)B(Z)}{C(Z)A(Z)}u(t) + \left(1 - \frac{D(Z)}{C(Z)}\right)y(t) \quad (2.59)$$

2.5.3 La forme matricielle

On peut écrire l'équation (2.57) par la forme matricielle suivant

$$y(t) = \hat{y}(t) + e(t) \quad (2.60)$$

Avec $\hat{y}(t) = \hat{y}(t)$: sortie estimée

$e(t) = y(t) - \hat{y}(t)$: erreur de prédiction

En utilisant les variables intermédiaires :

$$w(t) = \frac{B(Z)}{A(Z)}u(t) \quad (2.61)$$

$$v(t) = y(t) - w(t) \quad (2.62)$$

D'après l'équation (2.59) on a:

$$C(Z)(\hat{y}(t) - y(t)) = \frac{D(Z)B(Z)}{A(Z)}u(t) - D(Z)y(t) \quad (2.63)$$

On remplace $y(t) - \hat{y}(t)$ par $e(t)$ et $\frac{B(Z)}{A(Z)}u(t)$ par $w(t)$ on a :

$$C(Z)e(t) = -D(Z) [w(t) - y(t)] \quad (2.64)$$

On ajoute le terme $B(Z)u(t)$ dans le coté droit de l'équation (2.64), et on obtient :

$$C(Z)e(t) = D(Z)[y(t) - w(t)] - B(Z)u(t) + B(Z)u(t) \quad (2.65)$$

$$C(Z)e(t) = D(Z)[y(t) - w(t)] - B(Z)u(t) + B(Z)\frac{A(Z)}{A(Z)}u(t) \quad (2.66)$$

$$C(Z)e(t) = D(Z)v(t) - B(Z)u(t) + A(Z)w(t) \quad (2.67)$$

$$\left(1 + \sum_{i=1}^p c_i\right) e(t) = \left(1 + \sum_{i=1}^q d_i\right) v(t) + \left(1 + \sum_{i=1}^n a_i\right) w(t) - \sum_{i=1}^m b_i u(t) \quad (2.68)$$

$$e(t) - v(t) - w(t) = - \sum_{i=1}^p c_i e(t) + \sum_{i=1}^q d_i v(t) + \sum_{i=1}^n a_i w(t) - \sum_{i=1}^m b_i u(t) \quad (2.69)$$

On a : $y(t) = v(t) + w(t)$ et $\hat{y}(t) = y(t) - e(t)$

Alors l'équation (2.69) devient :

$$\hat{y}(t) = \sum_{i=1}^m b_i u(t) + \sum_{i=1}^p c_i e(t) - \sum_{i=1}^q d_i v(t) - \sum_{i=1}^n a_i w(t) \quad (2.70)$$

$$^T = [b_1 \dots b_m, c_1 \dots c_p, d_1 \dots d_q, a_1 \dots a_n]$$

$$^T(t) = [u(t) \dots u(t-m), e(t-1) \dots e(t-p), -v(t-1) \dots -v(t-q), \\ -w(t-1) \dots -w(t-n)]$$

Avec $w(t)$: Sortie sans bruit et $v(t)$:Bruit blanc filtrée.

D'après les équations de (2.7) à (2.20) on obtient l'algorithme suivant :

$$\begin{cases} \chi(t) = \chi(t-1) + P(t)(t) \left(y(t) - \mathcal{X}(t-1)(t) \right) & (2.71) \\ P(t) = p(t-1) - \frac{p(t-1)(t)^T(t)p(t-1)}{1 + ^T(t)p(t-1)(t)} & (2.72) \end{cases}$$

2.6 Conclusion

Dans ce chapitre on a présenté la méthode d'estimation des paramètres d'un système modélisé par les modèles stochastique suivants:

- Modèle ARMA au sens de la méthode des moindres carrés (récurives, pondérés).
- Modèle ARMAX au sens de la méthode des moindres carrés étendus.
- Modèle OE au sens de la méthode moindres carrés récurives.
- Modèle BJ au sens de la méthode moindres carrés récurives.

Ces méthodes permettent de minimiser l'erreur entre le système réel et le modèle identifié.

Le prochain chapitre on va faire des simulations afin de tester la fiabilité de ces algorithmes.

Chapitre 3 Simulation et interprétation des résultats

3.1 Introduction

Dans ce chapitre, on va faire une étude comparative entre des modèles stochastiques (ARX, ARMAX, BJ, OE), pour envisager les avantages et les inconvénients de ces modèles. Afin de tester les performances des algorithmes des moindres carrés vus précédemment, plusieurs tests de simulations doivent être effectués sur différents processus stables à phase minimale ou non minimale. Nous montrons par la suite, l'importance de la corrélation du bruit de mesure, la validité des résultats issus des hypothèses d'indépendance, ces simulations sont faites l'aide du logiciel de programmation MATLAB.

3.2 Bruit blanc

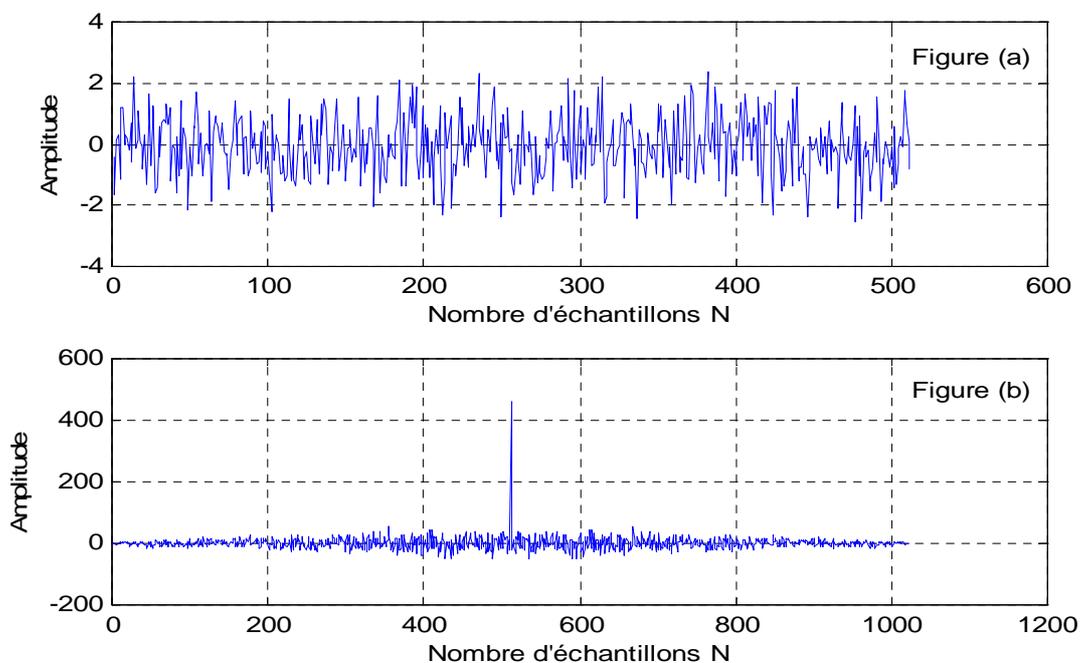


Figure 3.1. Signal du bruit $e(t)$ et sa fonction d'auto-corrélation.

3.3 Séquence d'entrée SBPA (Séquence Binaire Pseudo Aléatoire)

Pour la simulation, on utilise systématiquement comme signal d'entrée, la séquence binaire pseudo aléatoire SBPA qui est riche en fréquence et en même temps elle approche le bruit blanc discret.

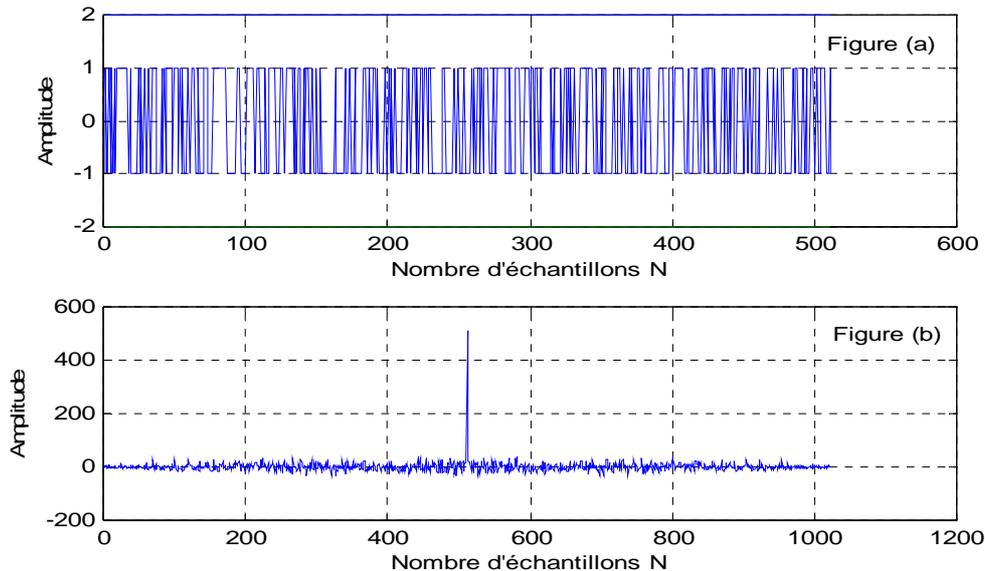


Figure 3.2. Séquence d'entrée SBPA en haut et sa fonction d'auto-corrélation en bas.

3.4 Algorithme des moindres carrés récursif (RLS)

Dans ce qui suit, nous allons effectuer des simulations sur différents processus pour tester la validité de l'algorithme des moindres carrés, nous allons générer plusieurs signaux avec des paramètres connus, et des ordres différents, pour lesquels nous allons appliquer cet algorithme dont le but de retrouver ces paramètres, et nous allons tester l'influence de nombre d'échantillons et de la variance du bruit additif sur l'évolution de ces paramètres.

3.4.1 Les processus de la simulation

Considérons un ensemble de quatre systèmes à simuler :

1^{er} système : Soit système physique stable à phase minimale d'ordre 1 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1}}{1 + 0.5z^{-1}} \text{ avec } a_1 = 0.5 \text{ et } b_1 = 1$$

2^{ème} système : Soit système physique stable à phase minimale d'ordre 2 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1} + 0.4z^{-2}}{1 + 0.5z^{-1} + 0.7z^{-2}} \text{ avec } a_1 = 0.5, a_2 = 0.7, b_1 = 1 \text{ et } b_2 = 0.4$$

3^{ème} système : Soit système physique stable à phase minimale d'ordre 3 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1} + 0.3z^{-2} + 0.5z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}}$$

avec $a_1 = 0.5, a_2 = 0.5, a_3 = 0.9, b_1 = 1, b_2 = 0.3 \text{ et } b_3 = 0.5$

4^{ème} système : Soit système physique stable à phase non minimale d'ordre 3 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1} + 2z^{-2} + 3z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}}$$

avec $a_1 = 0.5, a_2 = 0.5, a_3 = 0.9, b_1 = 1, b_2 = 2 \text{ et } b_3 = 3$

3.4.2 Influence du nombre d'échantillons

Nous allons tester l'évolution des paramètres de chaque système en agissant sur le nombre d'échantillons N avec la variance $\delta^2 = 0.0001$:

N \ \hat{a}_1, \hat{b}_1	\hat{a}_1	\hat{b}_1	Ecart type
256*1	0.5004	1.0001	0.0698
256*2	0.5011	1.0007	0.0490
256*3	0.5008	1.0001	0.0405
256*4	0.5002	1.0004	0.0351

Tableau 3.1. Influence du nombre d'échantillons pour le 1^{er} système.

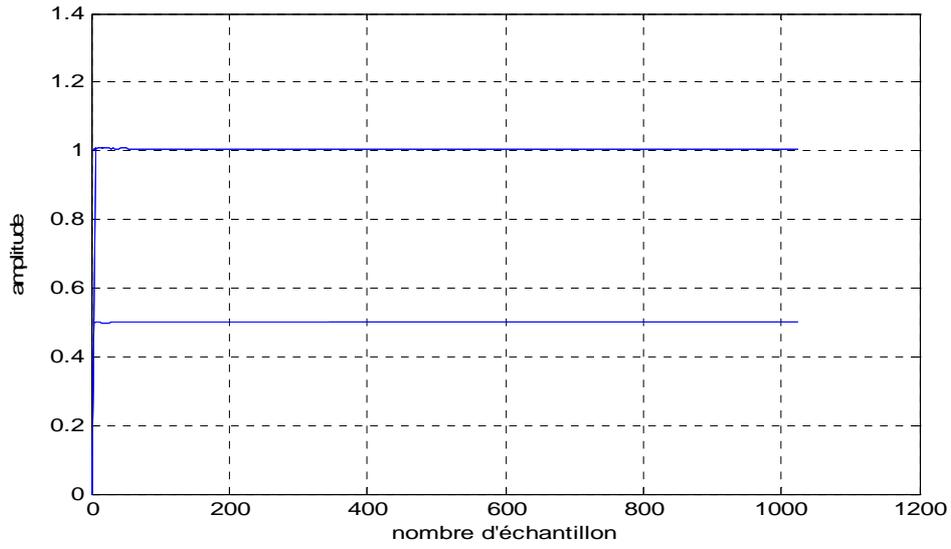


Figure 3.3. L'évolution des paramètres ($N = 256*4$, $\delta^2 = 0.0001$).

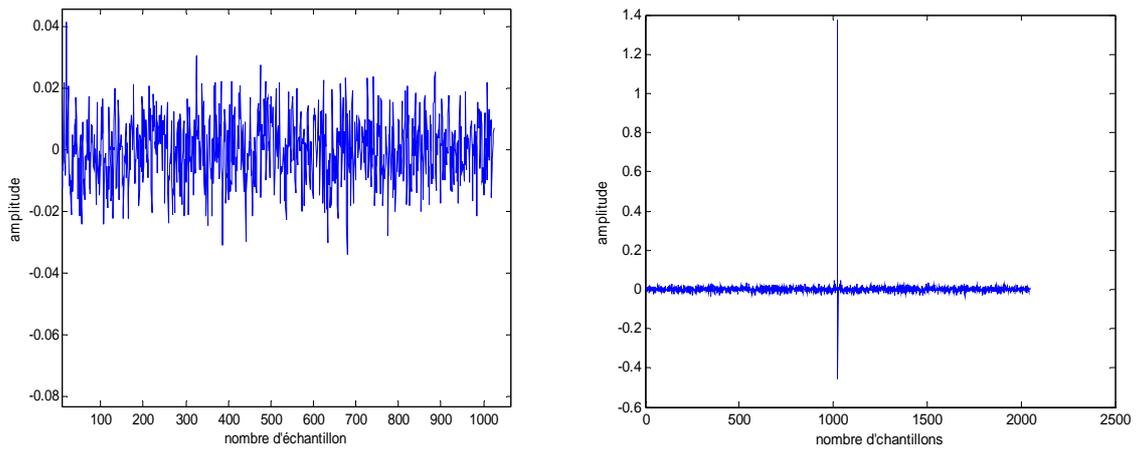


Figure 3.4. Erreur de prédiction et sa fonction d'auto-corrélation.

\hat{a}_i, \hat{b}_i	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	Ecart type
N					
256*1	0.4997	0.6994	1.0013	0.3990	0.0858
256*2	0.4997	0.6997	1.0009	0.3986	0.0608
256*3	0.4996	0.6996	1.0008	0.3986	0.0499
256*4	0.4998	0.6999	1.001	0.3992	0.0434

Tableau 3.2. Influence de nombre d'échantillons N pour le 2^{ème} système.

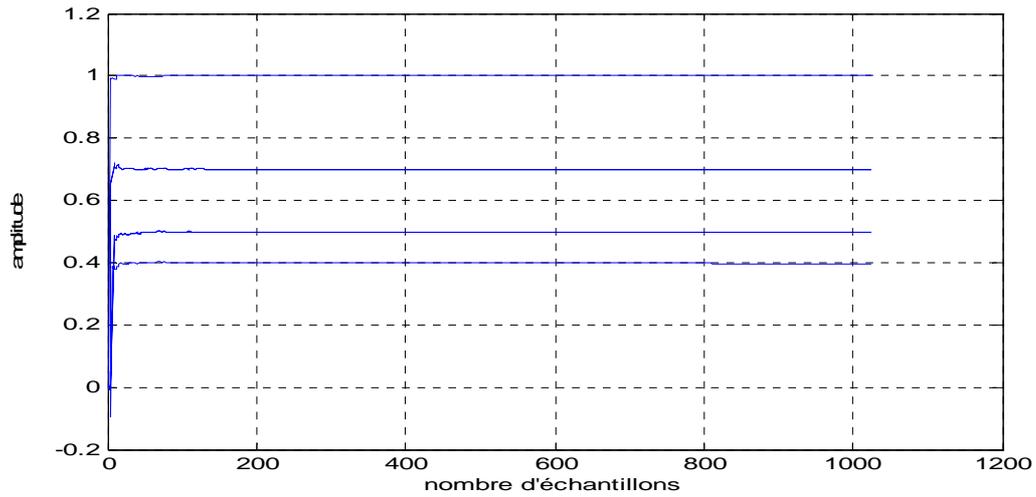


Figure 3.5. Evolution des paramètres ($N=1024$, $\delta^2=0.0001$)

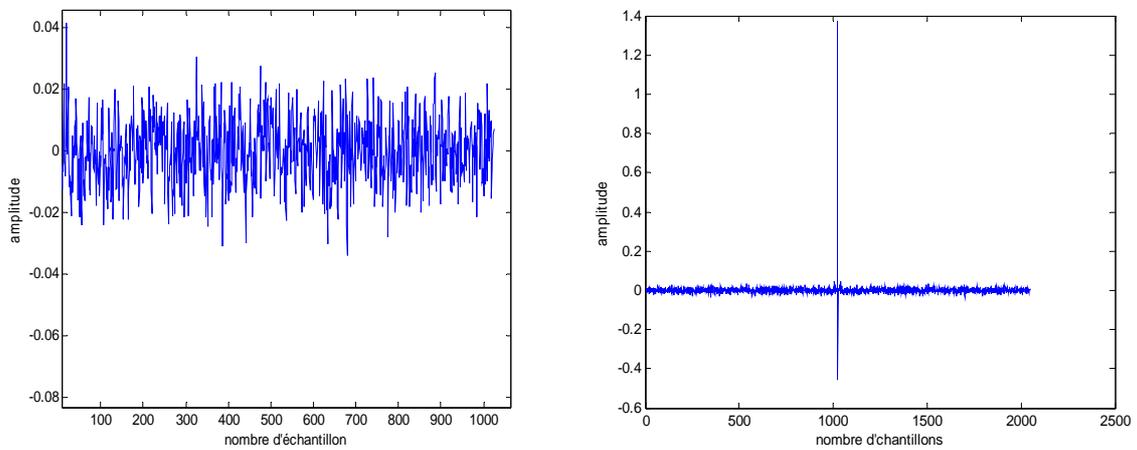


Figure 3.6. Le signal de l'erreur de prédiction et son auto corrélation.

\hat{a}_i, \hat{b}_i N	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
256*1	0.4970	0.5014	0.8971	1.0003	0.2983	0.4999	0.1264
256*2	0.4994	0.5003	0.8994	0.9999	0.2996	0.5002	0.0895
256*3	0.4992	0.5004	0.8992	0.9997	0.2996	0.5001	0.0734
256*4	0.5001	0.5000	0.9001	1.0001	0.2999	0.5005	0.0638

Tableau 3.3. Influence du nombre d'échantillons N pour le 3^{ème} système.

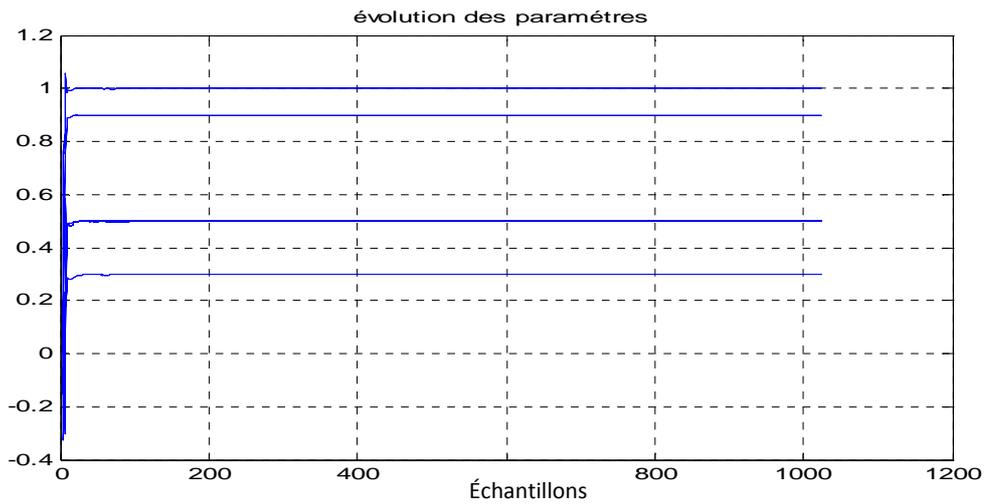


Figure 3.7. Evolution des paramètres ($N=1024$, $\delta^2=0.0001$)

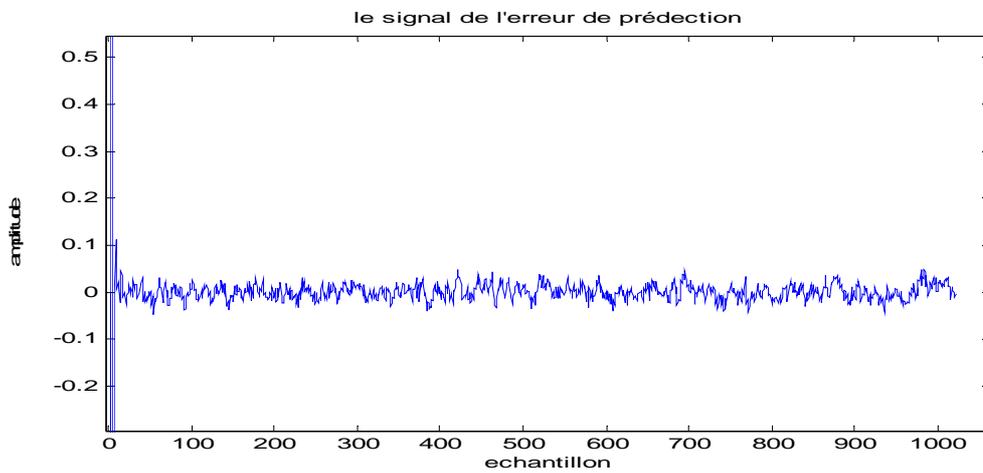


Figure 3.8. Signal de l'erreur de prédiction.

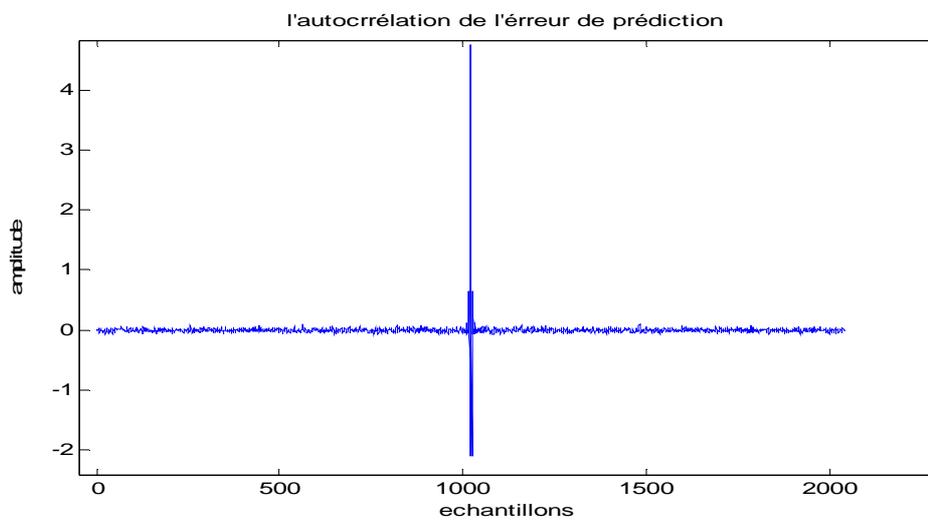


Figure 3.9. Fonction d'auto-corrélation de l'erreur.

\hat{a}_i, \hat{b}_i N	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
256*1	0.4994	0.5003	0.8993	0.9983	2.0013	3.0005	0.5658
256*2	0.4997	0.5001	0.8997	0.9986	2.0004	2.9995	0.3992
256*3	0.4996	0.5002	0.8996	0.9987	2.0002	2.9990	0.3258
256*4	0.4999	0.5000	0.8999	0.9994	2.0003	3.0001	0.2821

Tableau 3.4. Influence du nombre d'échantillons N pour le 4^{ème} système.

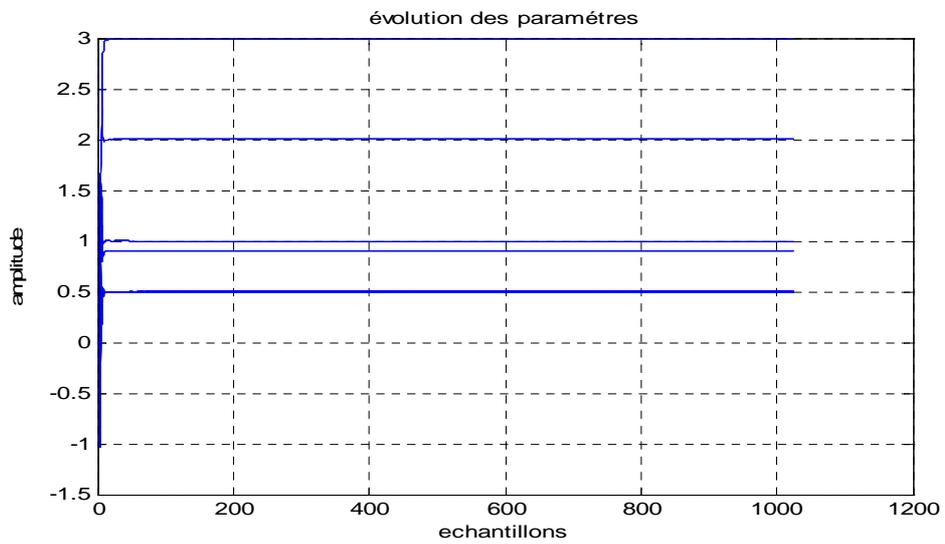


Figure 3.10. Evolution des paramètres (N=1024, $\delta^2=0.0001$)

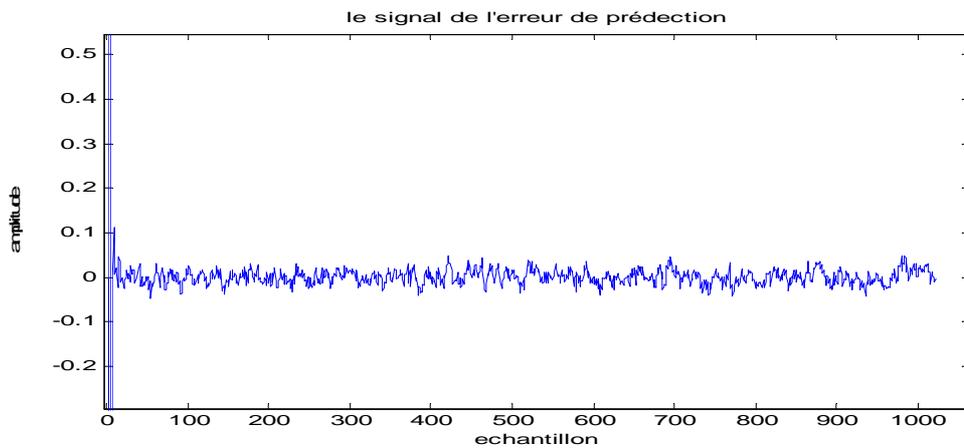


Figure 3.11. Erreur de prédiction.

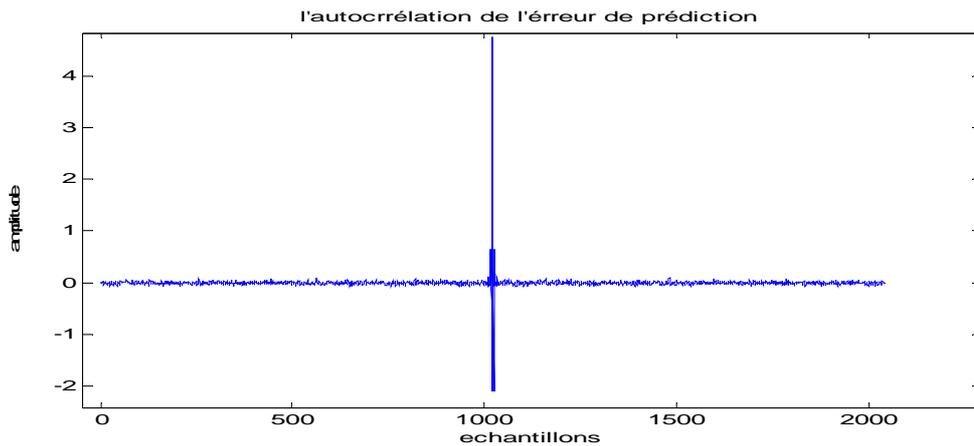


Figure 3.12. Fonction d’auto-corrélation de l’erreur.

Remarque : On remarque que lorsqu’on augmente le nombre d’échantillons N , l’écart type de l’erreur de prédiction diminue ce qui garantit une meilleure précision des paramètres estimés.

3.4.3 Influence de la variance δ^2

Nous allons tester l’évolution des paramètres de chaque système (voir Tableaux ci-dessous) en agissant sur la variance du bruit additif, et le nombre d’échantillons $N= 1024$.

$\delta^2 \backslash \hat{a}_i, \hat{b}_i$	\hat{a}_1	\hat{b}_1	Ecart type
1	0.1328	0.7779	1.1029
0.1	0.3828	0.9516	0.8170
0.01	0.4915	0.9973	0.0611
0.001	0.4988	0.9994	0.0307
0.0001	0.5000	1.002	0.0249

Tableau 3.5. Influence de la variance δ^2 pour le 1^{er} système.

$\delta^2 \backslash \hat{a}_i, \hat{b}_i$	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	Ecart type
1	0.1063	0.3436	1.0723	-0.2307	0.5641
0.1	0.4004	0.6173	1.0136	0.2636	0.2660
0.01	0.4907	0.6917	1.0034	0.3936	0.0998
0.001	0.4984	0.6991	1.0010	0.3964	0.0491
0.0001	0.5005	0.6998	0.9999	0.4010	0.0432

Tableau 3.6. Influence de la variance δ^2 pour le 2^{ème} système.

$\delta^2 \backslash \hat{a}_i, \hat{b}_i$	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
1	0.0125	0.5103	0.3649	0.9116	0.0296	0.2390	0.8173
0.1	0.2559	0.5839	0.6700	0.9001	0.2099	0.4064	0.3255
0.01	0.4348	0.5274	0.8390	0.9781	0.2724	0.4877	0.1396
0.001	0.4906	0.5042	0.8911	0.9978	0.2955	0.5016	0.0713
0.0001	0.4986	0.5005	0.8986	0.9987	0.2996	0.4989	0.0618

Tableau 3.7. Influence de la variance δ^2 pour le 3^{ème} système.

$\delta^2 \backslash \hat{a}_i, \hat{b}_i$	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
1	0.1395	0.6561	0.5584	0.8102	2.0457	2.0112	1.1076
0.1	0.3826	0.5545	0.7903	0.9654	1.9594	2.7023	0.4141
0.01	0.4805	0.5090	0.8819	0.9900	2.0038	2.9490	0.2963
0.001	0.4976	0.5011	0.8977	0.9977	2.0017	2.9928	0.2860
0.0001	0.5000	0.5000	0.9000	1.0000	2.0000	3.0001	0.2815

Tableau 3.8. Influence de la variance δ^2 pour le 4^{ème} système.

3.4.4 Interprétation

D'après les résultats de simulations présentés par des tableaux et des figures, on peut tirer les remarques suivantes :

- La précision des paramètres à estimer dépend à la fois de la variance et aussi du nombre d'échantillons. Lorsque le nombre d'échantillons est assez grand et la variance est faible, on aura une très bonne estimation des paramètres.
- Les notions de phase minimale (c.à.d. les zéros soient à l'intérieur du cercle unité) ou non minimale (c.à.d. les zéros soient à l'extérieur du cercle unité) n'influent pas sur la convergence de l'algorithme RLS, l'essentiel c'est que le système soit stable (c.à.d. les pôles soient à l'intérieur du cercle unité).
- L'auto-corrélation de l'erreur de prédiction de chaque système est proche à celle du bruit blanc, donc les modèles sont validés.

3.5 Algorithmes des moindres carrés pondérés (λ RLS)

3.5.1 Influence du facteur d'oubli λ

Dans cette partie, nous allons tester l'algorithme des moindres carrés pondérés λ RLS, en agissant sur le facteur d'oubli λ avec $N = 1024$ et $\delta^2 = 0.0001$:

λ \ \hat{a}_i, \hat{b}_i	\hat{a}_1	\hat{b}_1	Ecart type
0.6	0.5071	0.9953	0.0375
0.7	0.5057	0.9933	0.03690
0.9	0.5026	0.9941	0.0366
0.95	0.5023	0.9972	0.0365
0.98	0.5009	0.9990	0.0300

Tableau 3.9. Influence du facteur d'oubli pour le 1^{er} système.

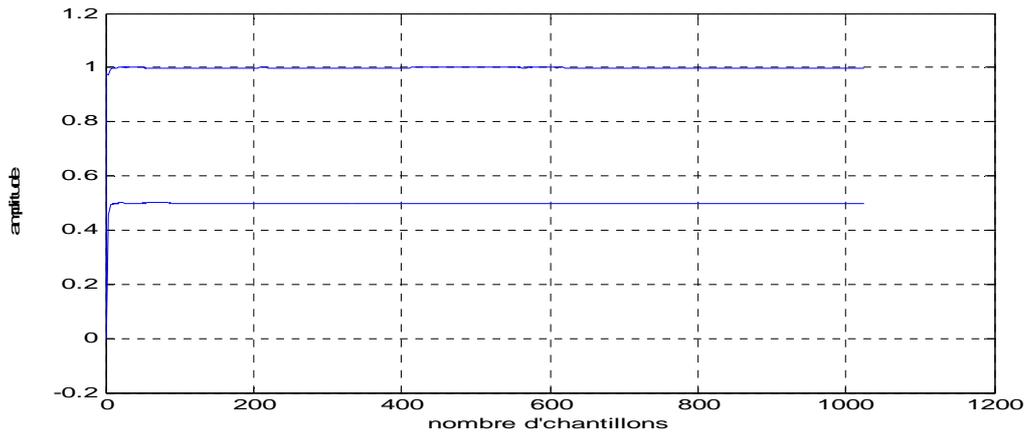


Figure 3.13. Evolution des paramètres ($N=1024$, $\delta^2=0.0001$, $\lambda=0.98$)

λ \ \hat{a}_i, \hat{b}_i	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	Ecart type
0.6	0.4856	0.6934	0.9901	0.3902	0.0474
0.7	0.5459	0.7196	0.9957	0.4639	0.0462
0.9	0.5073	0.6998	1.0065	0.4044	0.0450
0.95	0.4979	0.6995	1.0023	0.3930	0.0445
0.98	0.4986	0.6995	1.0019	0.3987	0.0300

Tableau 3.10. Influence de facteur d'oubli pour le 2^{ème} système.

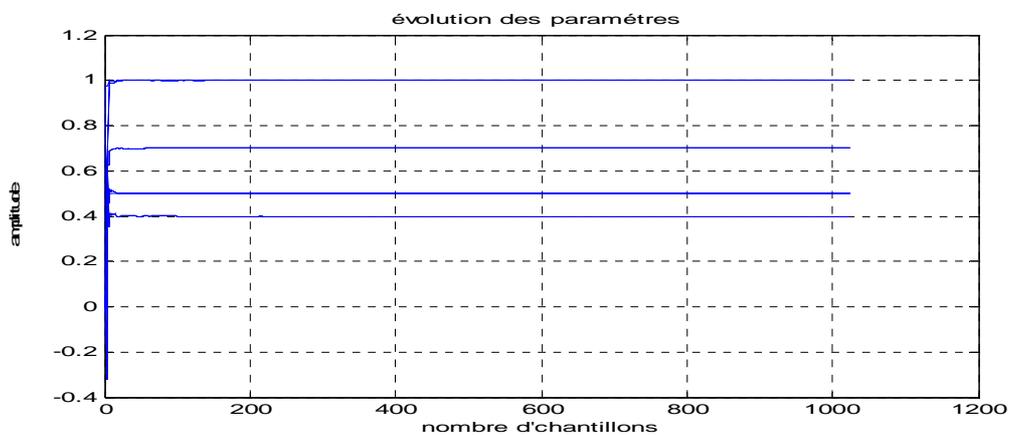


Figure 3.14. Evolution des paramètres ($N = 1024$, $\delta^2 = 0.0001$, $\lambda = 0.98$)

$\lambda \backslash \hat{a}_i, \hat{b}_i$	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
0.6	0.5175	0.5198	0.9008	0.9970	0.3036	0.5200	0.0684
0.7	0.5162	0.5171	0.9007	0.9971	0.3035	0.5178	0.0663
0.9	0.5042	0.5027	0.9001	0.9969	0.2975	0.5055	0.0641
0.95	0.4949	0.5030	0.8960	0.9959	0.2992	0.4971	0.0631
0.98	0.4998	0.4996	0.9000	1.0020	0.3014	0.4993	0.0622

Tableau 3.11. Influence du facteur d'oubli pour le 3^{ème} système

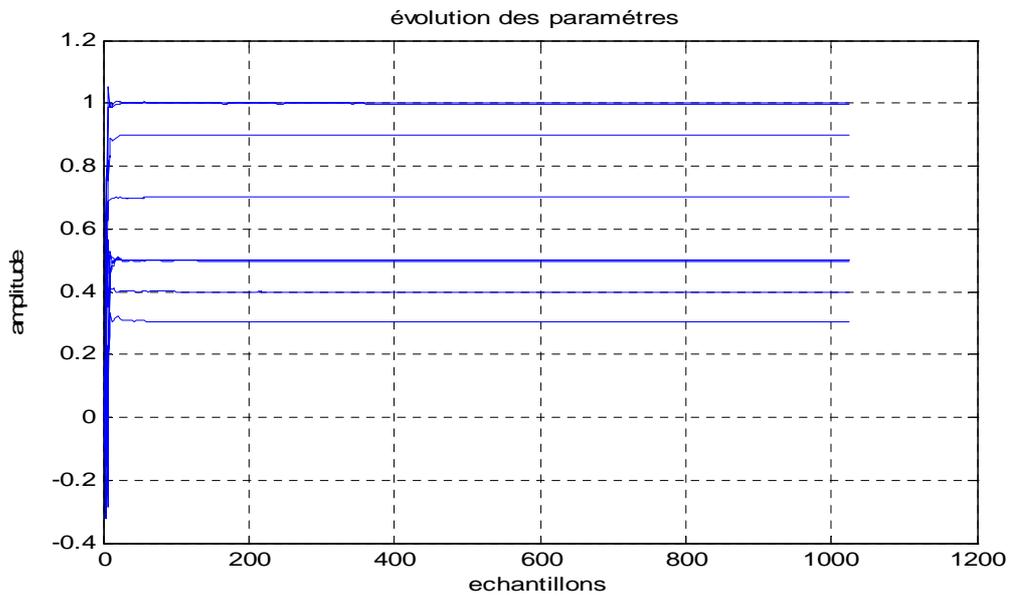


Figure 3.15. Evolution des paramètres (N=1024, $\delta^2=0.0001$, $\lambda=0.98$)

\hat{a}_i, \hat{b}_i λ	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	Ecart type
0.6	0.5016	0.5029	0.9027	1.0081	1.9928	3.0088	0.2810
0.7	0.5000	0.5012	0.9006	0.9948	2.0079	3.0062	0.2804
0.9	0.4969	0.5015	0.8971	0.9996	1.9998	2.9913	0.2800
0.95	0.5003	0.4997	0.9005	0.9960	2.0035	2.9978	0.2799
0.98	0.4999	0.4999	0.9001	1.0029	2.0010	2.9969	0.2683

Tableau 3.12. Influence de facteur d'oubli pour 4^{ème} système.

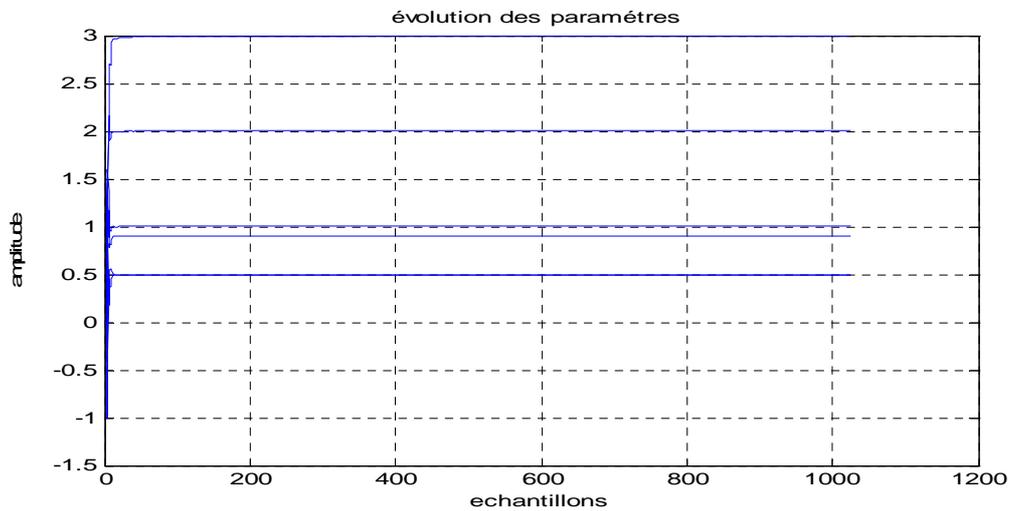


Figure 3.16. Evolution des paramètres (N=1024, $\delta^2=0.0001$, $\lambda=0.98$)

3.5.2 Interprétation

D'après les résultats obtenus, nous remarquons que l'algorithme λ RLS converge plus rapidement que l'algorithme RLS et donne une bonne estimation des paramètres dans le cas où λ varie entre 0.90 et 0.98. Donc il faut choisir une valeur de λ qui appartient à cet intervalle pour avoir une meilleure précision des paramètres.

3.6 Algorithme des moindres carrés étendus (RELS)

Dans ce qui suit, nous allons effectuer des simulations sur différents processus stables afin de tester les performances de l'algorithme des moindres carrés étendu. Nous allons générer plusieurs processus avec des paramètres connus, et des ordres différents, pour lesquels nous allons appliquer cet algorithme dont le but de retrouver ces paramètres. Nous allons tester par la suite l'influence du nombre d'échantillons et de la variance sur l'évolution de ces paramètres.

3.6.1 Les processus de la simulation

1^{ère} système : Considérons un système physique stable d'ordre 1 modélisé par un modèle ARMAX :

$$H_1(z) = \frac{z^{-1}}{1 + 0.5z^{-1}} \quad ; \quad H_2(z) = \frac{1 + 0.7z^{-1}}{1 + 0.5z^{-1}}$$

avec $a_1 = 0.5$, $b_1 = 1$, $c_1 = 0.7$

2^{ème} système : Considérons un système physique stable d'ordre 2 et à phase minimal modélisé par un modèle ARMAX :

$$H_1(z) = \frac{z^{-1} + 0.4z^{-2}}{1 + 0.5z^{-1} + 0.7z^{-2}} \quad ; \quad H_2(z) = \frac{1 + 0.1z^{-1} + 0.4z^{-2}}{1 + 0.5z^{-1} + 0.7z^{-2}}$$

avec $a_1 = 0.5$, $a_2 = 0.7$, $b_1 = 1$, $b_2 = 0.4$, $c_1 = 0.1$ et $c_2 = 0.4$.

3^{ème} système : Considérons un système physique stable d'ordre 3 et à phase minimal modélisé par un modèle ARMAX :

$$H_1(z) = \frac{z^{-1} + 0.3z^{-2} + 0.5z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}} \quad ; \quad H_2(z) = \frac{1 + 0.4z^{-1} + 0.1z^{-2} + 0.7z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}}$$

avec $a_1 = 0.5$, $a_2 = 0.5$, $a_3 = 0.9$, $b_1 = 1$, $b_2 = 0.3$, $b_3 = 0.5$, $c_1 = 0.4$, $c_2 = 0.1$ et $c_3 = 0.7$.

4^{ème} système : Considérons un système physique stable d'ordre 3, et à phase non minimal modélisé par un modèle ARMAX :

$$H_1(z) = \frac{z^{-1} + 2z^{-2} + 3z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}} \quad ; \quad H_2(z) = \frac{1 + 0.4z^{-1} + 0.1z^{-2} + 0.7z^{-3}}{1 + 0.5z^{-1} + 0.5z^{-2} + 0.9z^{-3}}$$

avec $a_1 = 0.5, a_2 = 0.5, a_3 = 0.9, b_1 = 1, b_2 = 2, b_3 = 3, c_1 = 0.4, c_2 = 0.1$ et $c_3 = 0.7$.

3.6.2 Influence du nombre d'échantillons

Nous allons tester l'évolution des paramètres de chaque système en agissant sur le nombre d'échantillons N.

N \ $\hat{a}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{b}_1	\hat{c}_1	Ecart type
256*1	0.5014	0.9996	0.6324	0.0761
256*2	0.4978	1.0010	0.6356	0.0570
256*3	0.4981	0.9994	0.6449	0.0516
256*4	0.4985	0.9997	0.6618	0.0479

Tableau 3.13. Influence du nombre d'échantillons 1^{er} système avec $\delta^2 = 0.001$.

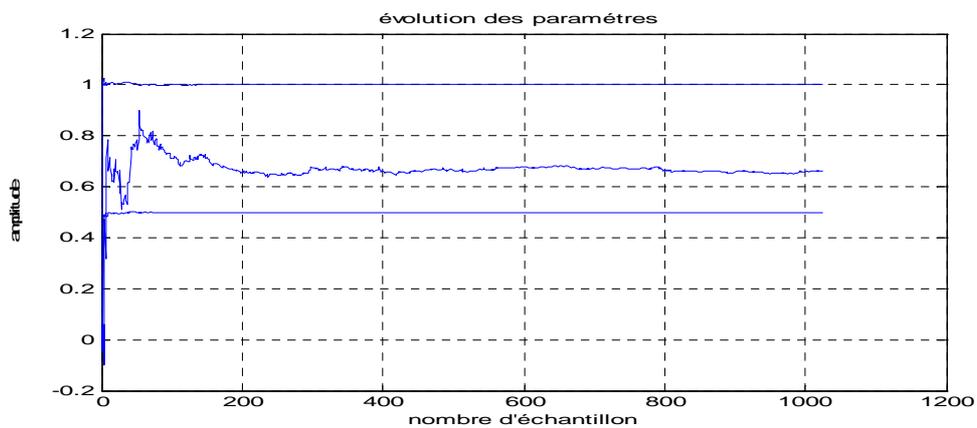


Figure 3.17. Evolution des paramètres (N = 1024, $\delta^2 = 0.001$).

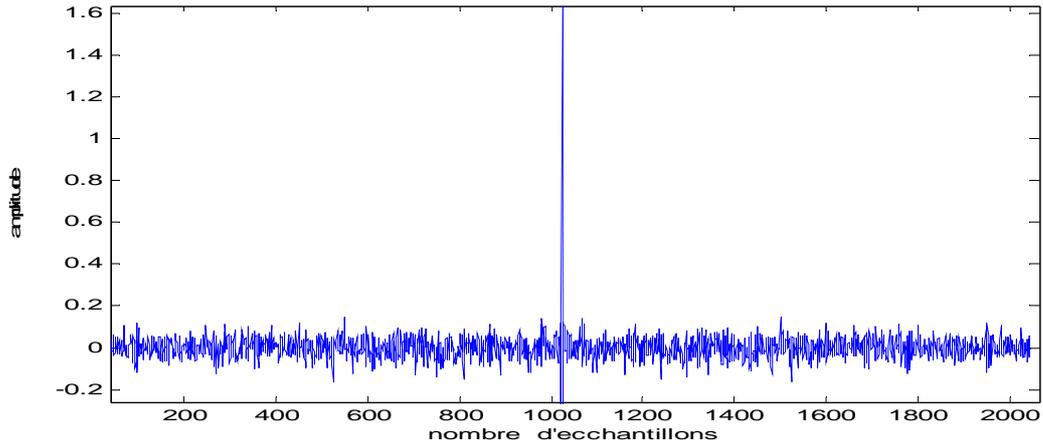


Figure 3.18. Auto-corrélation de l'erreur de prédiction

$\hat{a}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	\hat{c}_1	\hat{c}_2	Ecart type
N							
256	0.5003	0.6990	0.9996	0.3979	0.0003	0.2145	0.0945
256*2	0.5003	0.7040	1.0012	0.4005	0.1076	0.3170	0.0698
256*3	0.4964	0.6991	0.9996	0.3981	0.0296	0.3447	0.0586
256*4	0.5022	0.6974	0.9992	0.4023	0.0786	0.2390	0.0554

Tableau 3.14. Influence du nombre d'échantillons 2^{ème} système avec $\delta^2 = 0.001$

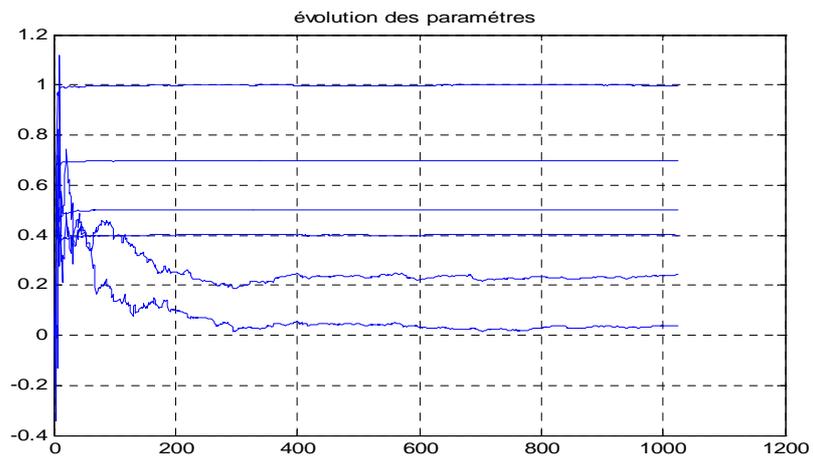


Figure 3.19. Evolution des paramètres ($N = 1024, \delta^2 = 0.001$).

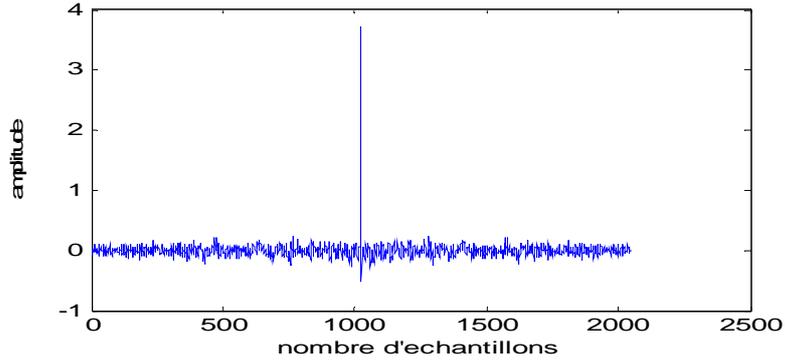


Figure 3.20. Auto-corrélation de l'erreur de prédiction.

$\hat{\alpha}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{c}_1	\hat{c}_2	\hat{c}_3	Ecart type
N										
256	0.4982	0.5025	0.9043	1.0019	0.2979	0.5042	0.3314	0.0594	0.6789	0.1369
256*2	0.5001	0.4989	0.8991	0.9987	0.2978	0.5002	0.4748	-0.0615	0.6527	0.1030
256*3	0.5001	0.5035	0.9040	0.9986	0.3053	0.5031	0.4702	-0.0460	0.6334	0.0848
256*4	0.4982	0.5007	0.9020	1.0008	0.2973	0.5023	0.3853	0.0200	0.6522	0.0753

Tableau 3.15. Influence du nombre d'échantillons $3^{\text{ème}}$ système avec $\delta^2 = 0.001$.

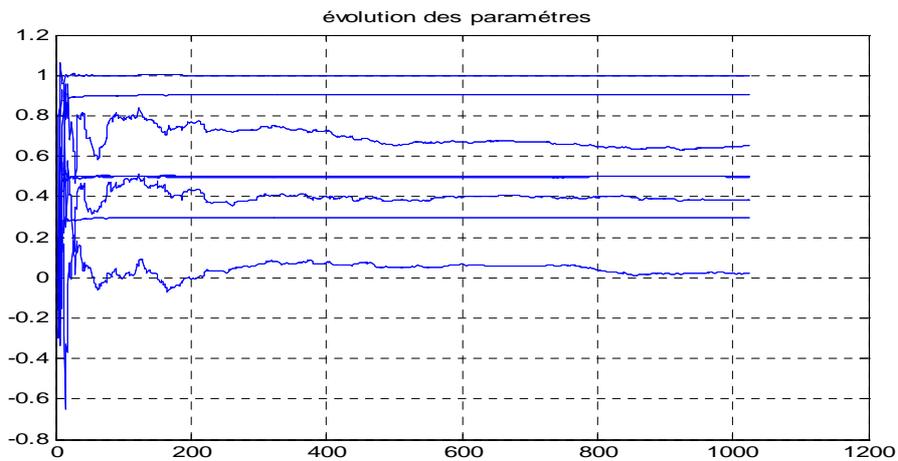


Figure 3.21. Evolution des paramètres ($N = 1024, \delta^2 = 0.001$).

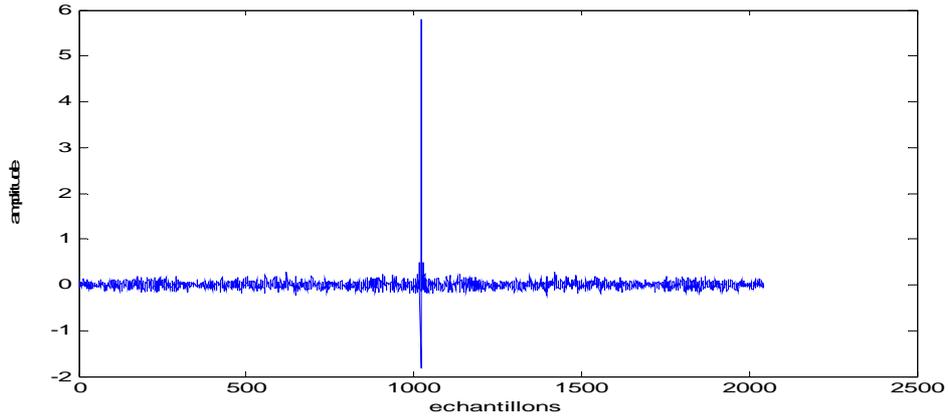


Figure 3.22. Auto-corrélation de l'erreur de prédiction.

$\hat{a}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{c}_1	\hat{c}_2	\hat{c}_3	Ecart type
N										
256	0.5000	0.5007	0.9004	0.9981	1.9998	3.0046	0.3524	0.0436	0.5399	0.5642
256*2	0.4997	0.5005	0.9002	0.9966	1.9985	3.0067	0.4349	0.0503	0.7058	0.3992
256*3	0.5002	0.5000	0.8998	0.9976	1.9983	3.0053	0.3869	-0.0520	0.5962	0.3352
256*4	0.4999	0.5000	0.9000	1.0018	1.9995	2.9998	0.4253	-0.0553	0.5835	0.2916
256*10	0.4998	0.4997	0.9000	0.9997	1.9979	3.0016	0.4536	-0.0457	0.6162	0.1806

Tableau 3.16. Influence du nombre d'échantillons pour le 4^{ème} système avec $\delta^2 = 0.001$.

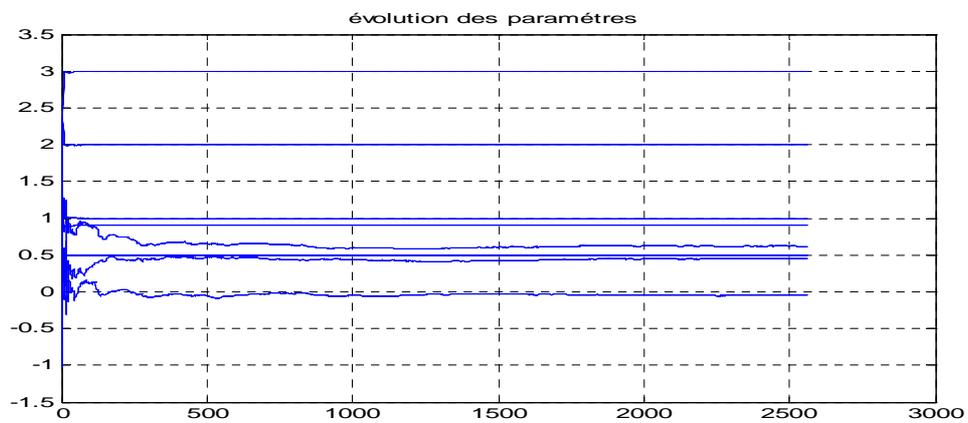


Figure 3.23. Evolution des paramètres ($N = 1024, \delta^2 = 0.001$).

3.6.3 Influence de la variance δ^2

Nous allons tester l'évolution des paramètres de chaque système en agissant sur la variance du bruit :

$\delta^2 \backslash \hat{a}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{b}_1	\hat{c}_1	Ecart type
1	0.4530	0.9334	0.5643	1.0092
0.1	0.4571	0.9679	0.6097	0.3445
0.01	0.5127	1.0075	0.6829	0.1088
0.001	0.4997	0.9999	0.6239	0.0489
0.0001	0.4987	0.9995	0.6324	0.0357

Tableau 3.17. Influence de la variance pour le 1^{er} système avec N=256*4.

$\delta^2 \backslash \hat{a}_i, \hat{b}_i, \hat{c}_i$	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	\hat{c}_1	\hat{c}_2	Ecart type
1	0.6687	0.7723	1.0928	0.5028	0.2495	0.4015	1.0383
0.1	0.4555	0.7139	0.9860	0.3904	-0.0476	0.2354	0.3362
0.01	0.5144	0.7100	1.0033	0.4120	0.0859	0.4020	0.1156
0.001	0.5025	0.6961	1.0048	0.3938	0.0882	0.3649	0.0549
0.0001	0.5009	0.7003	1.0000	0.4007	0.0435	0.2921	0.0446

Tableau 3.18. Influence de la variance pour le 2^{ème} système avec N=256*4.

$\hat{a}_i, \hat{b}_i, \hat{c}_i$ δ^2	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{c}_1	\hat{c}_2	\hat{c}_3	Ecart type
1	0.4708	0.4937	0.9303	0.9936	0.3844	0.3644	0.4147	0.0152	0.6987	1.1139
0.1	0.4874	0.5216	0.9357	1.0231	0.2600	0.5334	0.3874	0.0680	0.7097	0.3796
0.01	0.4953	0.4906	0.8989	0.9884	0.3151	0.4818	0.4232	-0.0739	0.5891	0.1542
0.001	0.4987	0.5001	0.9013	0.9992	0.2981	0.5034	0.3412	0.0013	0.6135	0.0751
0.0001	0.5003	0.5004	0.9001	0.9992	0.2997	0.5016	0.3270	0.0180	0.5744	0.0663

Tableau 3.19. Influence de la variance pour le 3^{ème} système avec N=256*4.

$\hat{a}_i, \hat{b}_i, \hat{c}_i$ δ^2	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{c}_1	\hat{c}_2	\hat{c}_3	Ecart type
1	0.4949	0.5090	0.9007	1.0671	1.9951	2.9846	0.4426	-0.0046	0.6690	1.2176
0.1	0.4956	0.5020	0.8975	1.0240	1.9492	3.0069	0.4533	-0.0548	0.6365	0.4625
0.01	0.5011	0.5000	0.8997	1.0039	2.0017	2.9967	0.3697	-0.0410	0.6003	0.2962
0.001	0.5000	0.5000	0.8998	1.0016	1.9992	3.0006	0.4651	-0.0276	0.6438	0.2862
0.0001	0.5002	0.5001	0.9001	0.9981	2.0007	3.0009	0.4473	-0.0474	0.5879	0.2813

Tableau 3.20. Influence de la variance pour le 4^{ème} système avec N=256*4.

3.6.4 Interprétation

D'après les résultats obtenus précédemment on peut affirmer que :

- Lorsque le nombre d'échantillons est assez grand et la variance est faible, on aura une très bonne estimation des paramètres du système.
- L'auto-corrélation des résidus des trois systèmes est proche à celle du bruit blanc donc les modèles sont validés.
- Les notions de phase minimale ou non minimale n'influent pas sur la convergence de l'algorithme(RELS), l'essentiel c'est que le système soit stable.

3.7 Algorithme de moindres carrée avec modèle OE (Output Error)

3.7.1 Les processus de la simulation

Considérons deux systèmes stables à phase minimale d'ordres 1 et 2 :

1^{er} système : Soit un système physique stable à phase minimale d'ordre 1 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1}}{1 + 0.5z^{-1}} \text{ avec } a_1 = 0.5, b_1 = 1$$

2^{ème} système : Soit système physique stable à phase minimale d'ordre 2 ayant la fonction de transfert suivante :

$$H(z) = \frac{z^{-1} + 0.4z^{-2}}{1 + 0.5z^{-1} + 0.7z^{-2}} \text{ avec } a_1 = 0.5, a_2 = 0.7, b_1 = 1 \text{ et } b_2 = 0.4$$

3.7.2 Influence du nombre d'échantillons

N \ \hat{a}_1, \hat{b}_1	\hat{a}_1	\hat{b}_1	Ecart type
256*1	0.5046	0.9958	0.0874
256*2	0.5037	0.9983	0.0602
256*3	0.5033	0.9987	0.0498
256*4	0.5018	0.9988	0.0438

Tableau 3.21. Influence du nombre d'échantillons pour le 1^{er} système avec $\delta^2 = 0.0001$.

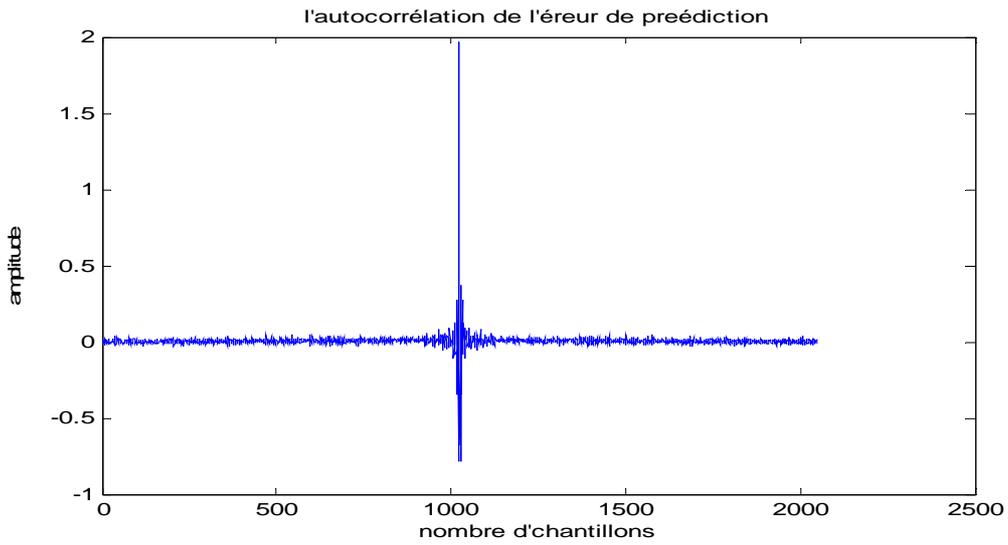


Figure 3.24. Auto-corrélation de l'erreur de prédiction.

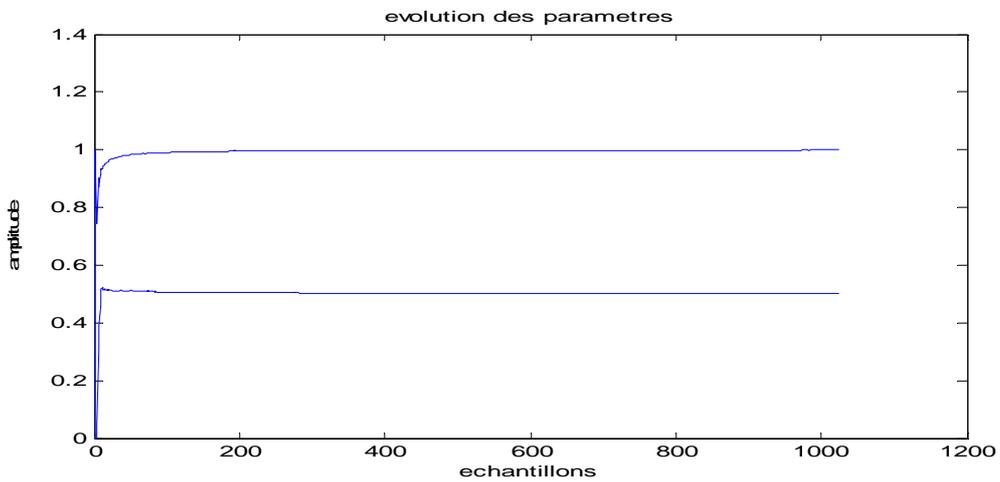


Figure 3.25. Evolution des paramètres ($N = 1024$, $\delta^2 = 0.0001$).

$N \backslash \hat{a}_1, \hat{b}_1$	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	Ecart type
256*1	0.5014	0.6707	0.9418	0.3533	0.4069
256*2	0.5037	0.6625	0.9522	0.3640	0.2854
256*3	0.5035	0.6609	0.9563	0.3692	0.2312
256*4	0.5015	0.6565	0.9612	0.3690	0.2034
256*8	0.5065	0.6580	0.9668	0.3786	0.1398

Tableau 3.22. Influence du nombre d'échantillons pour le 2^{ème} système avec $\delta^2 = 0.0001$.

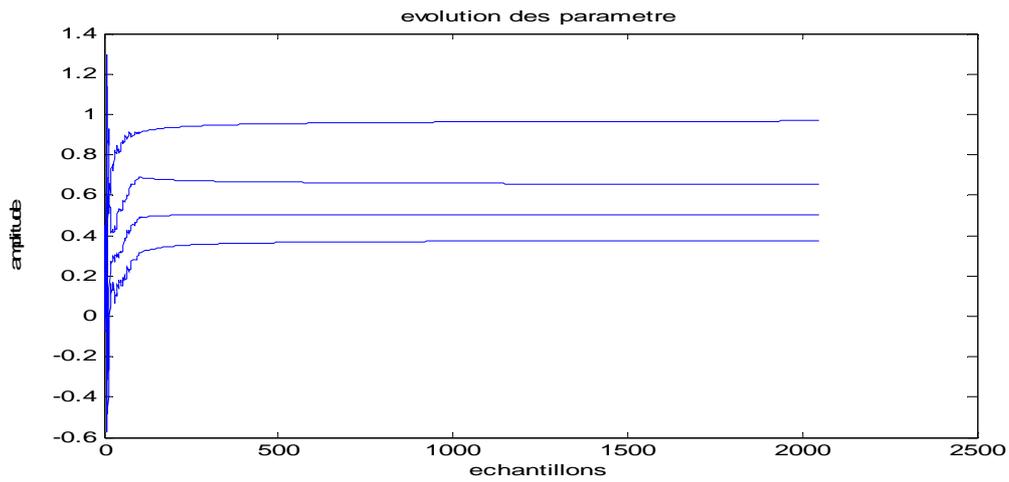


Figure 3.26. Evolution des paramètres ($N=1024$, $\delta^2=0.0001$)

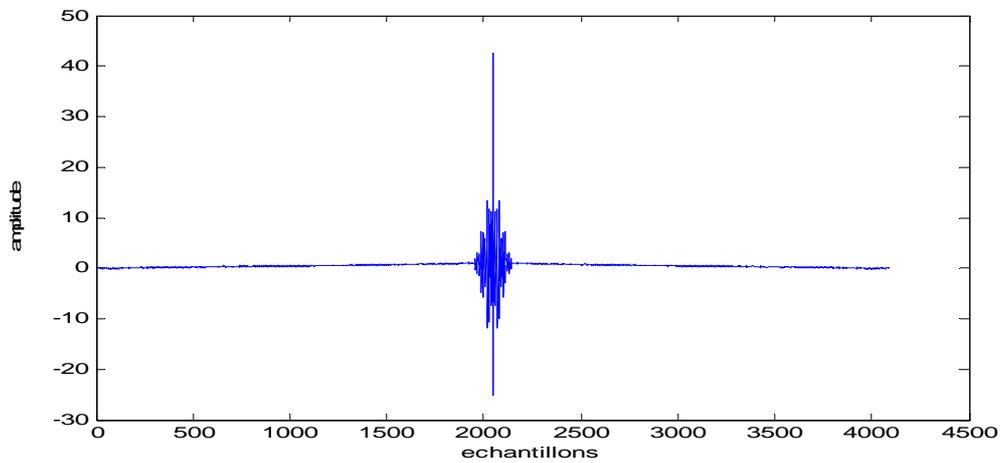


Figure 3.27. Auto-corrélation de l'erreur de prédiction

3.7.3 Influence de la variance

$\delta^2 \backslash \hat{a}_1, \hat{b}_1$	\hat{a}_1	\hat{b}_1	Ecart type
1	0.4478	0.9830	1.0140
0.1	0.5016	1.0039	0.3316
0.01	0.5099	1.0016	0.1100
0.001	0.4997	0.9984	0.0528
0.0001	0.5030	0.9996	0.0437

Tableau 3.23. Influence de la variance pour le 1^{er} système avec $N=256*4$.

$\delta^2 \backslash \hat{a}_1, \hat{b}_1$	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	Ecart type
1	0.5306	0.6201	0.9499	0.4385	1.0199
0.1	0.5639	0.6291	0.9469	0.4274	0.3686
0.01	0.4622	0.6338	0.9385	0.3221	0.2411
0.001	0.5005	0.6534	0.9599	0.3701	0.2061
0.0001	0.5032	0.6603	0.9619	0.3711	0.2011

Tableau 3.24. Influence de la variance pour le 2^{ème} système avec N=256*4.

3.7.4 Interprétation

D'après les résultats de simulations présentés par des tableaux et des figures ci-dessus, on peut conclure que :

- La précision des paramètres à estimer dépend de la variance et du nombre d'échantillons.
- L'auto-corrélation des résidus des deux systèmes n'est pas proche à celle du bruit blanc (les paramètres estimés ne convergent plus vers les paramètres réels), donc les modèles ne sont pas validés.

3.8 Algorithme de moindres carrée avec modèle BJ (Box-Jenkins)

3.8.1 Les processus de la simulation

Considérons deux systèmes à simuler :

1^{er} système : Soit système physique stable à phase minimale d'ordre 1 ayant les fonctions de transferts suivantes :

$$H_1(z) = \frac{z^{-1}}{1 + 0.5z^{-1}} \quad ; \quad H_2(z) = \frac{1 + 5z^{-1}}{1 + 0.8z^{-1}}$$

avec $a_1 = 0.5, b_1 = 1, c_1 = 5, d_1 = 0.8$

2^{ème} système : Soit système physique stable à phase minimale d'ordre 2 ayant les fonctions de transfert suivantes :

$$H_1(z) = \frac{z^{-1} + 0.4z^{-2}}{1 + 0.5z^{-1} + 0.7z^{-2}} \quad ; \quad H_2(z) = \frac{1 + 0.7z^{-1} + 0.3z^{-2}}{1 + 0.8z^{-1} + 0.2z^{-2}}$$

avec $a_1 = 0.5, a_2 = 0.7, b_1 = 1, b_2 = 0.4, c_1 = 0.7, c_2 = 0.3, d_1 = 0.8$ et $d_2 = 0.2$

3.8.2 Influence de nombre d'échantillons

$\hat{a}_i, \hat{b}_i, \hat{c}_i, \hat{d}_i$ N	\hat{a}_1	\hat{b}_1	\hat{c}_1	\hat{d}_1	Ecart type
256*1	0.5083	1.0050	4.5342	0.5465	0.0821
256*2	0.5016	1.0006	4.6966	0.6132	0.0535
256*3	0.5051	1.0032	4.8874	0.6969	0.0479
256*4	0.5137	1.0088	4.8440	0.6690	0.0461
256*8	0.5154	1.0100	5.0115	0.8098	0.0336

Tableau 3.25. Influence du nombre d'échantillons pour le 1^{er}système avec $\delta^2 = 0.0001$.

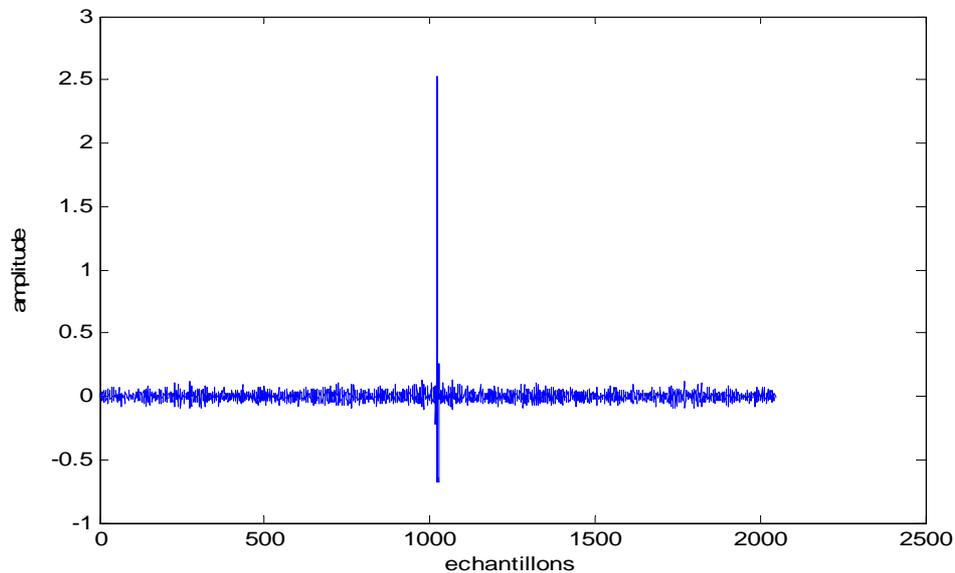


Figure 3.28. Auto-corrélation de l'erreur de prédiction.

$\hat{a}_1, \hat{b}_1, \hat{c}_1, \hat{d}_1$ N	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	\hat{c}_1	\hat{c}_2	\hat{d}_1	\hat{d}_2	Ecart type
256*1	0.4951	0.6981	0.9923	0.3975	0.3564	0.2077	0.5833	0.6856	0.1782
256*2	0.4964	0.6974	0.9941	0.4003	0.4581	0.8394	0.5750	0.6865	0.1292
256*3	0.4957	0.6980	0.9945	0.4002	0.6010	0.8653	0.5813	0.6810	0.1037
256*4	0.4939	0.6970	0.9959	0.3979	0.3042	0.7700	0.5674	0.6884	0.0928
256*8	0.4969	0.6979	0.9949	0.4014	0.4717	0.8760	0.5763	0.6907	0.0643

Tableau 3.26. Influence du nombre d'échantillons pour le 2^{ème} système avec $\delta^2 = 0.0001$.

3.8.3 Influence de la variance

$\hat{a}_1, \hat{b}_1, \hat{c}_1, \hat{d}_1$ δ^2	\hat{a}_1	\hat{b}_1	\hat{c}_1	\hat{d}_1	Ecart type
1	0.8022	1.1763	4.8286	0.6781	1.0336
0.1	0.7949	1.1883	5.0598	0.8579	0.3781
0.01	0.7189	1.1407	5.3104	1.1086	0.1430
0.001	0.5600	1.0391	5.2998	1.1461	0.0676
0.0001	0.5102	1.0066	4.8241	0.6983	0.0411

Tableau 3.27. Influence de la variance pour le 1^{er} système avec $N=256*4$.

$\hat{a}_1, \hat{b}_1, \hat{c}_1, \hat{d}_1$ δ^2	\hat{a}_1	\hat{a}_2	\hat{b}_1	\hat{b}_2	\hat{c}_1	\hat{c}_2	\hat{d}_1	\hat{d}_2	Ecart type
1	0.5130	0.4724	0.9337	0.3250	0.7942	0.4395	0.8158	0.3559	1.0138
0.1	0.4584	0.6457	0.9740	0.3675	0.5640	0.8388	0.6372	0.7193	0.3435
0.01	0.5015	0.6966	1.0019	0.3887	0.4894	0.8291	0.6157	0.6724	0.1373
0.001	0.4921	0.6962	0.9940	0.3961	0.4385	0.6983	0.5851	0.6753	0.0943
0.0001	0.4966	0.6975	0.9942	0.4009	0.4305	0.7616	0.5820	0.6853	0.0908

Tableau 3.28. Influence de la variance pour le 2^{ème} système avec $N=256*4$.

3.8.4 Interprétation

D'après les résultats des tableaux et des figures, on peut dire que :

- La précision des paramètres à estimer dépend de la variance et aussi du nombre d'échantillons.
- L'auto-corrélation des résidus des deux systèmes n'est pas proche de celle du bruit blanc (les paramètres estimés ne convergent pas vers les paramètres réels), donc les modèles ne sont pas validés.

3.9 Conclusion

Dans ce chapitre, nous avons implémenté sous MATLAB les algorithmes d'identification paramétrique basés sur le blanchissement de l'erreur de prédiction, qui sont : l'algorithme des moindres carrés récursif (RLS), l'algorithme des moindres carrés pondéré (λ RLS), l'algorithme des moindres carrés étendus (RELS) et l'algorithme des moindres carrés basés sur le modèle d'erreur de sortie (Output Error) et l'algorithme des moindres carrés basés sur le modèle (Box-Jenkins).

D'après plusieurs tests de simulations par les algorithmes (RLS, RELS), on a remarqué que l'erreur tend asymptotiquement vers un bruit blanc, ce qui garantit une estimation non biaisée des paramètres d'un système physique stable.

Par contre, la simulation avec les algorithmes des moindres carrés appliqués aux modèles (OE et BJ), nous donne une estimation biaisée des paramètres pour un système physique stable.

Chapitre 4 Application au signal de parole

4.1 Introduction

L'analyse de la parole est une étape indispensable à toute application de synthèse, de codage, ou de reconnaissance. Elle repose en général sur un modèle. Celui-ci possède un ensemble de paramètres numériques, dont les plages de variation définissent l'ensemble des signaux couverts par le modèle. Pour un signal et un modèle donné, l'analyse consiste en l'estimation des paramètres du modèle dans le but de lui faire correspondre le signal analysé. Pour ce faire, on met en œuvre un algorithme d'analyse, qui cherche généralement à minimiser la différence, appelée erreur de modélisation, entre le signal original et celui qui serait produit par le modèle

Dans ce chapitre, nous allons faire une application de la méthode des moindres carrés au signal de la parole bruitée, mais avant de présenter les résultats de simulations, nous présenterons quelques généralités sur le traitement de la parole.

4.2 Le mécanisme de phonation

Le signal vocal est le résultat de plusieurs actions conjuguées. En effet, la parole résulte de l'action coordonnée des appareils respiratoires et masticatoires contrôlés par le système nerveux central. L'appareil respiratoire fournit l'air qui est expiré par la trachée artère. L'air passe ensuite par le larynx où sa pression est modulée grâce aux cordes vocales qui déterminent la taille de l'ouverture (la glotte) par laquelle il peut passer. Finalement, l'air transite par le conduit vocal qui s'étend du pharynx aux lèvres pour devenir le signal vocal qui est émis par le locuteur. La Figure 4.1 représente un schéma général de l'appareil phonatoire humain. On peut

distinguer deux grandes classes de sons émis par l'appareil phonatoire : les sons voisés et les sons non voisés.

- Les sons voisés, de forte énergie, résultent de l'excitation du conduit vocal par des impulsions périodiques de pression liées aux oscillations des cordes vocales. À chaque impulsion la glotte s'ouvre brusquement et libère la pression accumulée en amont puis elle se referme plus graduellement.
- Les sons non voisés, beaucoup moins énergétiques que les sons voisés, résultent quant à eux de l'écoulement libre de l'air par la glotte qui reste ouverte. Le chuchotement est un mode de phonation particulier car la glotte reste ouverte en permanence, il n'y a donc pas de production de sons voisés pendant le chuchotement.

Il est important de remarquer que le signal vocal n'est pas stationnaire, son évolution suit les déformations du conduit vocal. Cependant, ces déformations sont suffisamment lentes pour que le signal vocal puisse être considéré comme stationnaire sur des périodes allant de 20 ms à 40 ms. On considère donc ce signal comme quasi-stationnaire sur de tels intervalles ce qui motive le choix du traitement par trames (de 20 ms à 40 ms) du signal vocal pour de nombreuses applications, notamment pour la réduction de bruit et le codage de la parole [15].

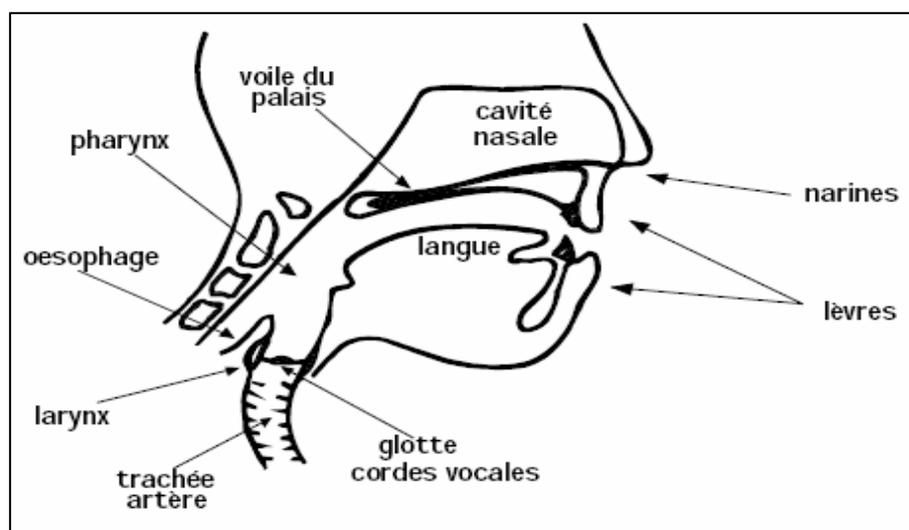


Figure 4.1. Représentation de l'appareil phonatoire humain.

4.3 Caractéristiques du signal de parole

L'analyse spectrale des sons voisés et non voisés donne des informations fondamentales dans le cadre du traitement de la parole et en particulier pour la réduction de bruit. Nous allons donc illustrer les principales caractéristiques des sons voisés et non voisés [15].

4.3.1 Les sons voisés

La hauteur d'un son voisé est liée à la fréquence de vibration des cordes vocales. Cette fréquence est appelée fréquence fondamentale ou pitch. Un son voisé est par définition un signal quasi-périodique qui possède un spectre fréquentiel très caractéristique, Deux sons de même intensité et de même hauteur se distinguent par le timbre qui est déterminé par les amplitudes relatives des harmoniques du fondamental. La fréquence fondamentale (F_0) peut varier [15] :

- de 80 à 200 Hz pour une voix masculine,
- de 150 à 450 Hz pour une voix féminine,
- de 200 à 600 Hz pour une voix d'enfant.

4.3.2 Les sons non voisés

Un son non voisé ne présente pas de structure périodique, il peut être considéré comme un bruit blanc filtré par le conduit vocal. Son spectre ne possède donc pas de structure particulière et c'est souvent dans les hautes fréquences que le spectre est le plus énergétique. [15]

4.4 Analyse et paramétrisation de la parole

Le signal de parole est un processus aléatoire non-stationnaire à long terme, mais il est considéré comme stationnaire dans des fenêtres temporelles d'analyse de l'ordre de 20 ms à 30 ms. Cette propriété de stationnarité à court terme permet donc une analyse et modélisation progressive du signal de parole accompagnée, bien

sur, d'un chevauchement de fenêtres pour permettre une continuité temporelle des caractéristiques de l'analyse et du modèle.

Le modèle généralement admis pour le signal de parole a une structure de modèle Autorégressif AR.

$$y(t) = -\sum_{i=1}^n a_i y(t-i) + e(t) \quad (4.1)$$

avec

n : Ordre du modèle

a_i : Coefficients de prédiction

$e(t)$: Bruit blanc (erreur sur le modèle).

4.5 Résultat de simulations

Considérons le signal de sortie comme un signal de parole qui est sous forme d'un fichier de données correspond à la phrase suivante : « un loup s'est jeté immédiatement sur la petite chèvre » la fréquence d'échantillonnage de ce signal et le nombre d'échantillons sont [16] :

$F_e=16\text{KHZ}$; $N=256*256$

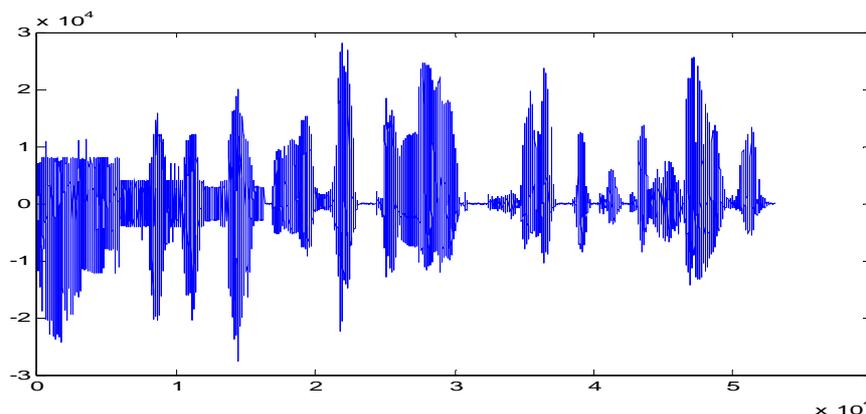


Figure 4.2. Le signal de parole

L'identification des paramètres du modèle AR correspondant au signal de parole, ne peut faire que sur un signal stationnaire, c'est pourquoi nous allons découper notre signal en des tranches de 30 ms.

Pour la simulation on a choisi deux trames de 512 échantillons illustrées sur les figures ci-dessous :

a) Partie voisée du signal de parole

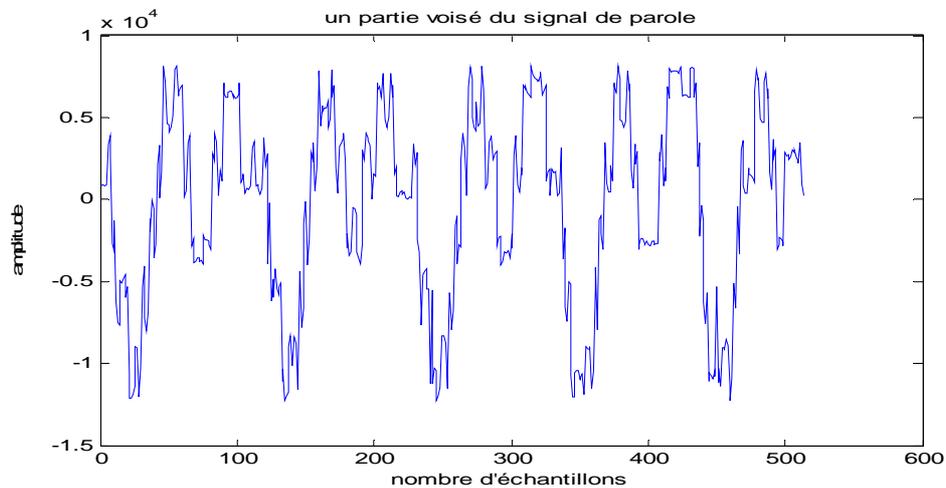


Figure 4.3. Une partie voisée du signal de parole

- **Résultats de simulations obtenus par l’algorithme RLS**

L’utilisation de l’algorithme des moindres carrés récursif RLS (Recursive Least Squares) pour identifier le paramètre a_i du modèle AR d’ordre 10, donne les résultats suivants :

Paramètres	Algorithme
a_i	RLS
a_1	-0.8108
a_2	-0.0631
a_3	0.0304
a_4	-0.1295
a_5	-0.0064
a_6	-0.0350
a_7	-0.0450
a_8	0.0508
a_9	0.0330
a_{10}	0.0700

Tableau 4.1. Estimation des paramètres du signal de parole.

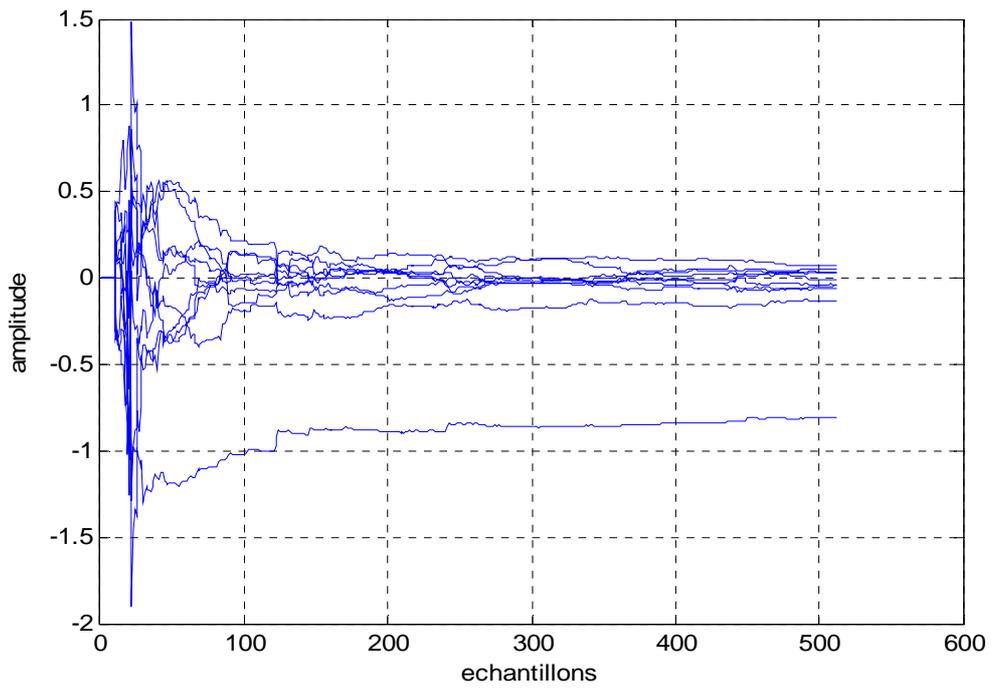


Figure 4.4. Evolution des paramètres du signal de parole.

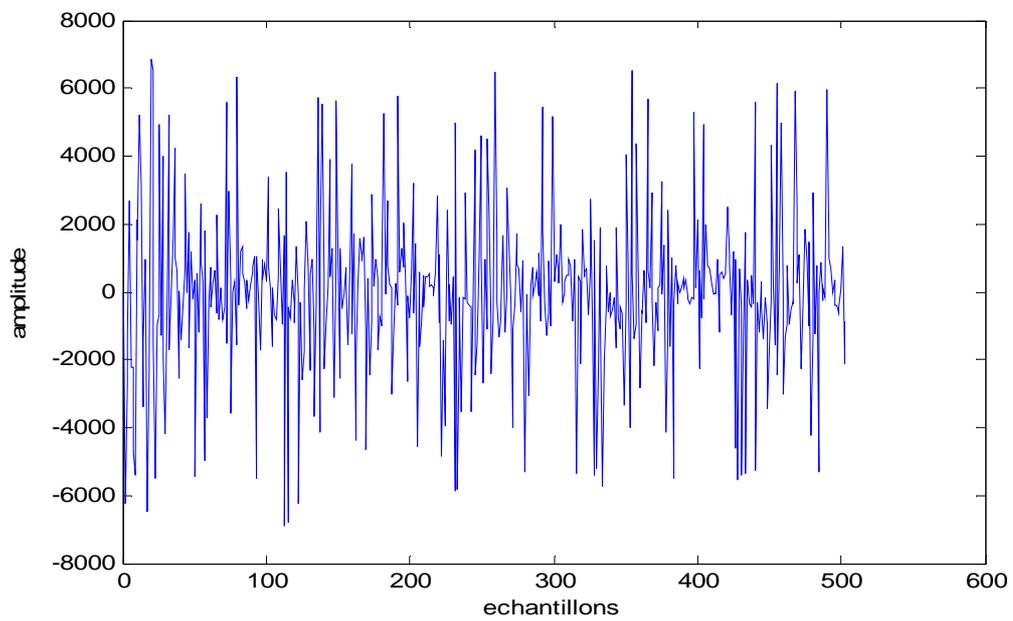


Figure 4.5. Le signal de l'erreur de prédiction.

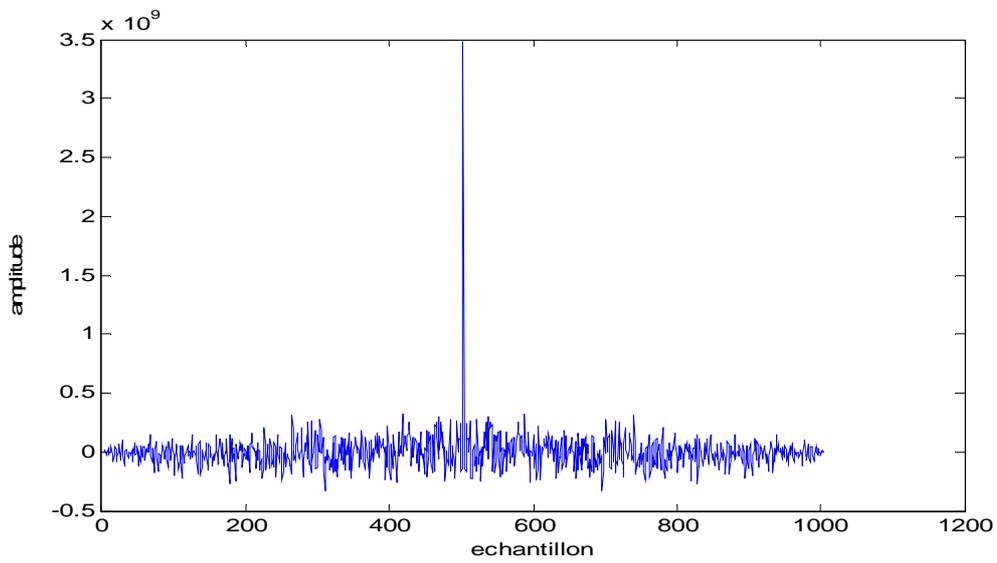


Figure 4.6. Auto corrélation de l'erreur de prédiction.

4.5.1 Partie non voisée du signal de parole(RLS)

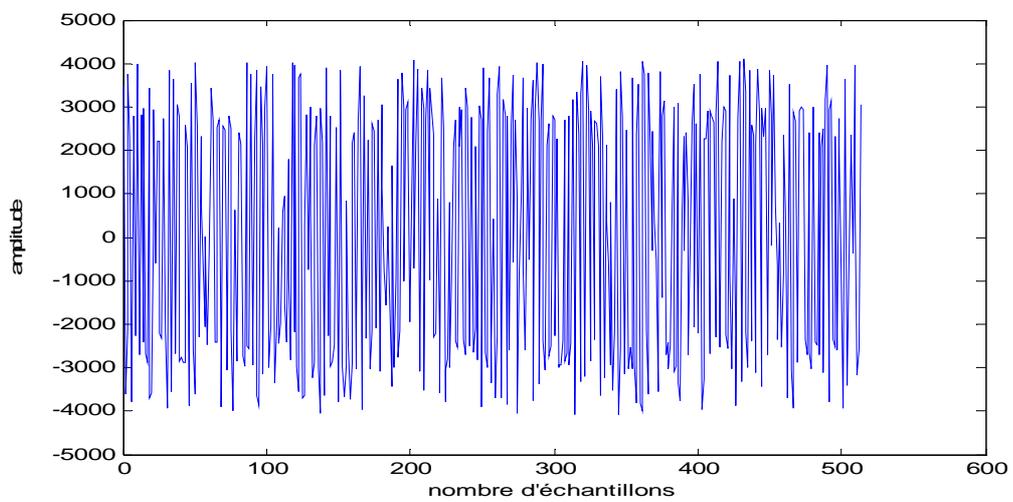


Figure 4.7. Une partie non voisée du signal de parole

- **Résultats de simulations obtenus par l'algorithme RLS**

L'utilisation de l'algorithme des moindres carrés récursif RLS (Recursive least square) pour identifier le paramètre a_i du modèle AR d'ordre 10, donne les résultats suivants :

Paramètres a_i	Algorithme RLS
a_1	0.0571
a_2	0.4354
a_3	-0.1637
a_4	-0.0579
a_5	-0.0572
a_6	-0.0264
a_7	-0.1063
a_8	0.0106
a_9	-0.0121
a_{10}	0.0164

Tableau 4.2. Estimation des paramètres du signal de parole.

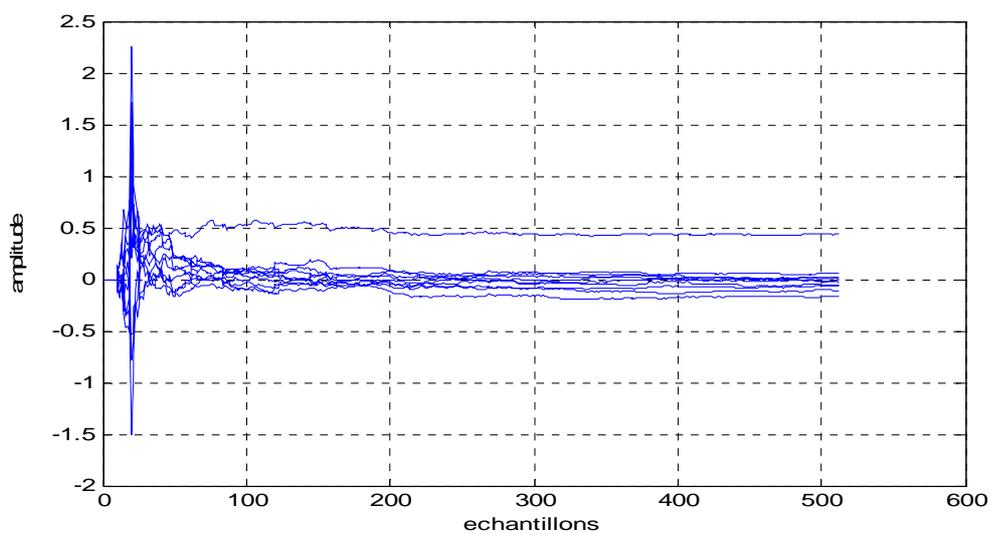


Figure 4.8. Evolution des paramètres du signal de parole.

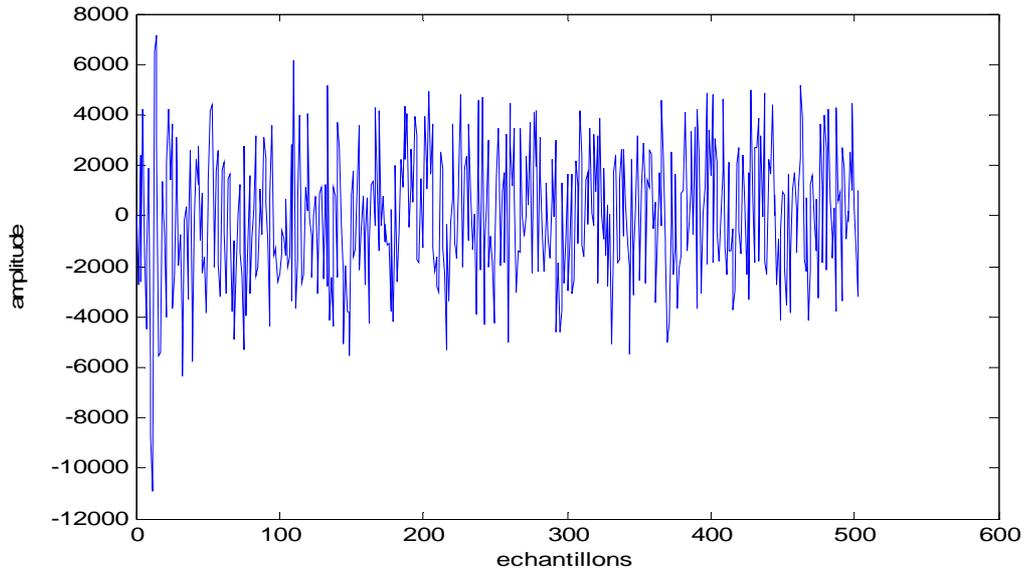


Figure 4.9. Signal de l'erreur de prédiction.

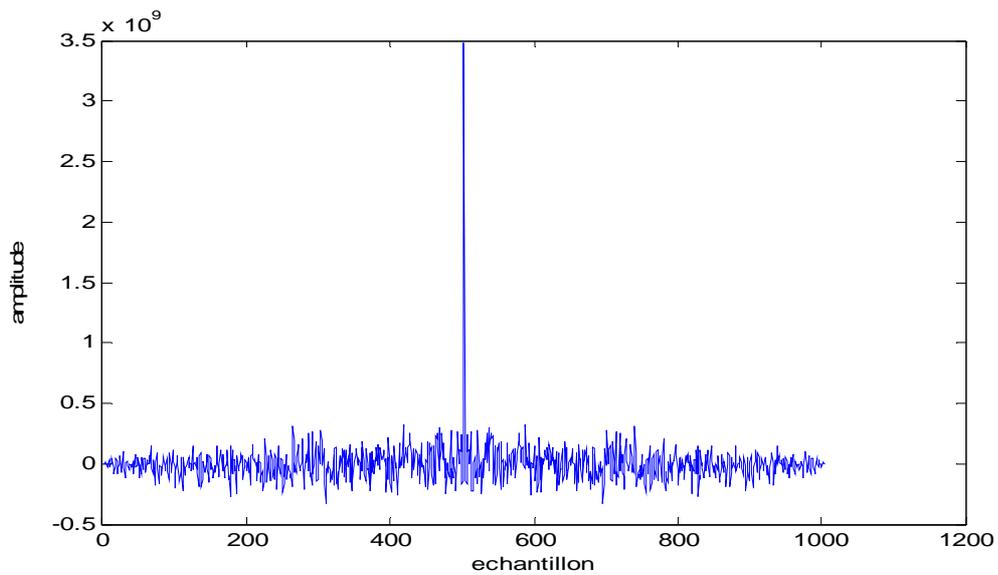


Figure 4.10. Auto corrélation de l'erreur de prédiction

4.5.2 Résultat de simulations par l'algorithme RELS

L'utilisation de l'algorithme des moindres carrés récursif RELS (Recursive Extended least squares) pour identifier les paramètres du modèle AR, sachant que nous avons ajouté du bruit à nos échantillons et cela pour différentes valeurs de SNR. Les résultats obtenus sont illustrés sur le tableau ci-dessous :

SNR \ a_i, c_i	5	10	20	50	80	100	paramètres sans bruit
a_1	0.0645	0.1540	0.0750	0.0449	0.0339	0.0294	0.0571
a_2	0.3231	0.2910	0.4507	0.4150	0.3845	0.3875	0.4354
a_3	-0.1740	-0.1381	-0.1546	-0.1740	-0.1808	-0.1833	-0.1637
a_4	-0.1242	-0.1487	-0.0600	-0.0670	-0.0767	-0.0741	-0.0579
a_5	-0.0323	-0.0438	-0.0647	-0.0591	-0.0507	-0.0510	-0.0572
a_6	-0.0138	-0.0369	-0.0359	-0.0269	-0.0238	-0.0232	-0.0264
a_7	-0.0977	-0.1038	-0.1110	-0.1042	-0.1026	-0.1016	-0.1063
a_8	0.0195	0.0041	0.0010	0.0126	0.0134	0.0128	0.0106
a_9	-0.0232	-0.0045	-0.0196	-0.0104	-0.0099	-0.0106	-0.0121
a_{10}	0.0210	0.0231	0.0168	0.0183	0.0155	0.0144	0.0164
c_1	0.0248	0.0932	0.0187	-0.0133	-0.0229	-0.0273	
c_2	-0.0735	-0.1367	0.0179	-0.0134	-0.04572	-0.0412	

Tableau 4.3. Estimation des paramètres d'un signal de parole en présence du bruit. (Partie non voisé)

Remarque :

Nous avons remarqué que si le signal de parole est plus bruité, l'estimation des paramètres est biaisée et la qualité de l'estimateur se dégrade rapidement lorsque le SNR diminue .

4.6 Conclusion

L'identification des paramètres du signal de parole par l'algorithme RLS donne de bons résultats, par contre l'identification des paramètres du signal bruité par l'algorithme RELS donne des résultats acceptables dans des conditions de bruit faible.

Conclusion générale

Dans ce travail de mémoire, nous avons étudié et simulé les méthodes d'identification paramétrique basées sur le blanchissement de l'erreur de prédiction afin d'obtenir des estimations des paramètres des systèmes non biaisés.

Les algorithmes d'identification développés sont :

- Algorithme des moindres carrés récursif (RLS)
- Algorithme des moindres carrés avec facteur d'oubli (λ RLS)
- Algorithme des moindres carrés étendu (RELS)
- Algorithme des moindres carrés appliqué au modèle Output Error (OE)
- Algorithme des moindres carrés appliqué au modèle Box-Jenkins (BJ)

Nous avons appliqué en premier lieu ces algorithmes sur des exemples de systèmes à paramètres connus, les résultats obtenus montrent que lorsque le nombre d'échantillons est assez grand et la variance est faible, on aura une bonne estimation des paramètres du système. Et d'après plusieurs tests de simulations par les algorithmes (RLS, RELS), on a remarqué que l'erreur tend asymptotiquement vers un bruit blanc, ce qui garantit une estimation non biaisée des paramètres d'un système physique stable. Par contre, la simulation avec les algorithmes des moindres carrés appliqués aux modèles (OE et BJ), nous donne une estimation biaisée des paramètres pour un système physique stable.

Ensuite, l'application au signal de parole, donne de très bon résultat par l'algorithme RLS appliqué sur un signal de parole modélisé par un modèle AR, par contre, l'identification des paramètres par l'algorithme RELS donne des résultats acceptables dans des conditions de parole faiblement bruitée, mais ces performances

se dégradent rapidement en présence de bruit. Dans la représentation temps fréquence du signal bruité, certains des paramètres de la parole sont masqués par le bruit. Dans un tel cas les paramètres du signal propre ne peuvent être estimés à partir du signal bruité et sont donc considérés comme manquants ou incertains.

Finalement, cette étude nous a permis d'avoir des connaissances sur la modélisation paramétrique et l'identification des systèmes physiques et mettre des premiers pas pour améliorer les méthodes de traitement du signal.

Parmi les perspectives de ce travail, nous souhaitons améliorer la qualité d'estimation des paramètres du signal de parole par la méthode de vrai semblance ainsi que leur implémentation sur le système d'identification Toolbox de MATLAB. Un second objectif porte sur l'extension des méthodes précédentes au cas des modèles non linéaires dans un contexte bruité. Finalement, Nous espérons que notre travail va enrichir la recherche du domaine d'identification des processus physiques.

Bibliographie

- [1] M. Kunt, <<traitement numérique des signaux>>, édition Dunod
- [2] Jaques Max et Jean-Louis Lacoume, <<méthodes et techniques de traitement du signal>>, 5ème édition Dunod, Paris 2000
- [3] M. Boumendil, M. Kadri, <<Etude et implémentation des paramètres et application sur un signal de parole>>, mémoire d'ingénieur d'état en électronique, Blida, octobre 2002.
- [4] A. Boumezoud, H. Salhi <<identification paramétrique par les méthodes basées sur le blanchissement de l'erreur de prédiction et application au signal de parole>>, thèse d'ingénieur d'état, Blida, juin, 2004.
- [5] Ioan D. Landau, <<identification des systèmes>>, HERMES, Paris, 1998.
- [6] Y. Oussar, <<Réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique de processus>>. thèse de Doctorat de l'université Pierre et Marie Curie – Paris VI 1998.
- [7] J. Richelet, G. Lavielle, J. Mallet, <<Commande prédictive Note de cours n°8 : Modélisation>>, l'ENSICA, France, http://artic.acbesancon.fr/Reseau_stl/CIRA/Commande%20pr%C3%A9dictive/note8.pdf
- [8] M. dehri, FZ. boudani, <<Etude et implémentation de la Méthode des Moindres carrés étendue : application au signal de parole bruité>>. Mémoire d'ingénieur d'état en électronique, Blida, juillet 2010.
- [9] Eric Walter et Luc Pronzato, <<Identification de modèle paramétrique à de données expérimentales>>, Masson
- [10] A. Beveniste, M. Metiver et P. Priouret, <<Algorithmes adaptatifs et approximation stochastiques>>. Masson., 1987.
- [11] M. Najim, <<Modélisation et identification en traitement du signal>>, Masson, 1988
- [12] D. Urbani, <<Méthodes statistiques de sélection d'architectures neuronales : application à la conception de modèles de processus dynamiques>>. thèse de Doctorat de l'université Pierre et Marie Curie – Paris VI, 1995.
- [13] A. Quinquis, <<Le traitement du signal sous matlab>>, Paris, 2000.
- [14] G. Binet, <<la méthode des moindres carrés en présence de bruit>>, UFR de Sciences, Universités de Caen, 2005, http://www.greyc.ensicaen.fr/~gbinet/Adaptatif/LesMC_bruit.pdf.
- [15] Cyril Plapous, <<traitements pour la réduction de bruit. Application à la communication parlée>>, thèse de doctorat université de Rennes 1 - IRISA/ENSSAT 12/12/2005.
- [16] A. Maddi, A. Guessoum, D. Berkani and O. Belkina, "Etude de la méthode des moindres carrés récursive et application au signal de parole", 3rd International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, SETIT 2005, Sousse, Tunisia, March 17-21, 2005.