



Modèles de volterra à complexité réduite: estimation paramétrique et application à l'égalisation des canaux de communication

Alain Kibangou

► To cite this version:

Alain Kibangou. Modèles de volterra à complexité réduite: estimation paramétrique et application à l'égalisation des canaux de communication. Automatique / Robotique. Université Nice Sophia Antipolis, 2005. Français. NNT: . tel-00190985

HAL Id: tel-00190985

<https://theses.hal.science/tel-00190985>

Submitted on 23 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université de Nice-Sophia Antipolis – UFR Sciences
École Doctorale des Sciences & Technologies de l'Information et de la
Communication

T H È S E

pour obtenir le titre de
DOCTEUR EN SCIENCES
de l'UNIVERSITE de Nice-Sophia Antipolis

Discipline : Automatique, Traitement du Signal et des Images

présentée et soutenue par
Alain Y. KIBANGOU

MODELES DE VOLTERRA A COMPLEXITE REDUITE : ESTIMATION PARAMETRIQUE ET APPLICATION A L'EGALISATION DES CANAUX DE COMMUNICATION

Thèse dirigée par **Gérard FAVIER et Moha M. HASSANI**
soutenue le 28 janvier 2005

Jury :

M. Francis CASTANIE	Professeur, TéSA, ENSEEIHT	Président
M. José RAGOT	Professeur, CRAN,I.N.P.L.	Rapporteur
M. Driss ABOUTAJDINE	Professeur,GSCM, Université Mohamed V	Rapporteur
M. Gérard FAVIER	Directeur de recherche au CNRS,I3S,UNSA	Directeur de thèse
M. Moha M. HASSANI	Professeur, LEI,Université Cadi Ayyad	Directeur de thèse

A
*Bertille, mon épouse,
Michèle Antoinette, ma mère,
Yvette, ma sœur aînée
et à toute la fratrie des KIBANGOU.*

Remerciements

Ce travail a été régi par une convention de co-tutelle de thèse entre l'université de Nice Sophia Antipolis(UNSA) et l'université Cadi Ayyad de Marrakech (UCAM). En conséquence, il a été conjointement préparé au sein du projet ASTRE du laboratoire I3S/CNRS/UNSA, à Sophia Antipolis, et au Laboratoire d'Electronique et Instrumentation (LEI) de la Faculté des Sciences Semlalia Marrakech (FSSM).

D'une manière particulière, je remercie :

- M. Gérard FAVIER pour m'avoir accueilli au sein de son équipe, et m'avoir fait profiter de son expérience de recherche et de sa rigueur à travers la direction de cette thèse. Je lui suis également très reconnaissant pour son soutien financier au travers de différents contrats.
- M. Moha M'rabet HASSANI pour m'avoir accueilli au sein du LEI, pour son encadrement et ses encouragements. Je lui suis également reconnaissant pour m'avoir représenté auprès des autorités administratives de l'UCAM pendant mon séjour en France.
- M. Francis CASTANIE d'avoir accepté de présider le jury de ma thèse.
- M. José RAGOT et M. Driss ABOUTAJDINE d'avoir accepté d'être rapporteurs de ce travail et pour leurs remarques constructives.

Mes sincères et profonds remerciements sont adressés aux collègues, amis et personnels du laboratoire I3S et de la FSSM, qui, directement ou non, ont contribué à la réalisation de ce travail.

Un grand merci à A. KHOUAJA, M. BOULOUIRD, A. DE ALMEIDA, E. FERNANDES, L. ROTA, V. ZARZOSO, M. RAJIH, K. BEN-CHEHIDA, O. PONSINI, E. WOLSTZINSKI, B. KHADDOUMI, C. LE DUC, S. BOUDAUD, S. KALOUN, E. ROCCHI, etc. pour leur amitié, leur gentillesse et les discussions constructives de tous ordres.

Ma profonde gratitude va aussi à l'endroit de ma famille et de tous mes amis pour le soutien moral qu'ils m'ont témoigné.

Un grand merci à Bertille pour son soutien sans faille et pour sa patience.

Que Dieu, source de ma vie, soit béni au travers de ce travail !

Résumé

Une large classe de systèmes physiques peut être représentée à l'aide du modèle de Volterra. Il a notamment été montré que tout système non-linéaire, invariant dans le temps et à mémoire évanouissante peut être représenté par un modèle de Volterra de mémoire et d'ordre finis. Ce modèle est donc particulièrement attrayant pour les besoins de modélisation et d'identification de systèmes non-linéaires.

Un des atouts majeurs du modèle de Volterra est la linéarité par rapport à ses paramètres, c'est à dire les coefficients de ses noyaux. Cette caractéristique permet d'étendre à ce modèle certains résultats établis pour l'identification des modèles linéaires. Il est à noter que le modèle de Volterra peut, par ailleurs, être vu comme une extension naturelle de la notion de réponse impulsionnelle des systèmes linéaires aux systèmes non-linéaires. Toutefois, certaines limitations sont à circonvier : un nombre de paramètres qui peut être très élevé et un mauvais conditionnement de la matrice des moments de l'entrée intervenant dans l'estimation du modèle au sens de l'erreur quadratique moyenne minimale (EQMM). Il est à noter que ce mauvais conditionnement est aussi à l'origine de la lenteur de convergence des algorithmes adaptatifs de type LMS (Least Mean Squares). Cette thèse traite principalement de ces deux questions. Les solutions apportées sont essentiellement basées sur la notion d'orthogonalité. D'une part, l'orthogonalité est envisagée vis à vis de la structure du modèle en développant les noyaux de Volterra sur une base orthogonale de fonctions rationnelles. Ce développement est d'autant plus parcimonieux que la base est bien choisie. Pour ce faire, nous avons développé de nouveaux outils d'optimisation des bases de Laguerre et BFOR (Base de Fonctions Orthonormales Rationnelles) pour la représentation des noyaux de Volterra. D'autre part, l'orthogonalité est envisagée en rapport avec les signaux d'entrée. En exploitant les propriétés statistiques de l'entrée, des bases de polynômes orthogonaux multivariés ont été construites. Les paramètres du modèle de Volterra développé sur de telles bases sont alors estimés sans aucune inversion matricielle, ce qui simplifie significativement l'estimation paramétrique au sens EQMM. L'orthogonalisation des signaux d'entrée a aussi été envisagée via une procédure de Gram-Schmidt. Dans un contexte adaptatif, il en résulte une accélération de la convergence des algorithmes de type LMS sans un surcoût de calcul excessif.

Certains systèmes physiques peuvent être représentés à l'aide d'un modèle de Volterra simplifié, à faible complexité paramétrique, tel que le modèle de Hammerstein et celui de Wiener. C'est le cas d'un canal de communication représentant l'accès à un réseau

sans fil via une fibre optique. Nous montrons notamment que les liaisons montante et descendante de ce canal peuvent respectivement être représentées par un modèle de Wiener et par un modèle de Hammerstein. Dans le cas mono-capteur, en utilisant un précodage de la séquence d'entrée, nous développons une solution permettant de réaliser l'estimation conjointe du canal de transmission et des symboles transmis de manière semi-aveugle. Il est à noter que, dans le cas de la liaison montante, une configuration multi-capteurs peut aussi être envisagée. Pour une telle configuration, grâce à un précodage spécifique de la séquence d'entrée, nous exploitons la diversité spatiale introduite par les capteurs et la diversité temporelle de sorte à obtenir une représentation tensorielle du signal reçu. En appliquant la technique de décomposition tensorielle dite PARAFAC, nous réalisons l'estimation conjointe du canal et des symboles émis de manière aveugle.

Mots clés : Modélisation, Identification, Bases orthogonales, Base de Laguerre, Base de fonctions orthonormales rationnelles, Polynômes orthogonaux, Optimisation de pôles, Réduction de complexité, Egalisation, Modèle de Volterra, Modèle de Wiener, Modèle de Hammerstein, Décomposition PARAFAC.

Abstract

A broad class of physical systems can be represented using the Volterra model. Particularly, it was shown that a truncated Volterra model could represent any non-linear system, time-invariant with fading memory. This model is thus particularly attractive for non-linear systems modeling and identification purpose.

One of the main advantages of the Volterra model is its linearity-in-parameters, i.e. the kernel coefficients. This property allows the extension of some results established for linear model identification to this model. In addition, one can note that the Volterra model can be viewed as the natural extension of the impulse response concept of linear systems to non-linear systems. However, some limitations are to be circumvented : the number of parameters which can be very high and the ill conditioning of the matrix of the input moments used in the model estimation in the Minimal Mean Square Error (MMSE) sense. It should be noted that this ill conditioning is also the origin of the convergence slowness of the LMS (Least Mean Squares). This thesis mainly deals with these two issues. The proposed solutions are primarily based on the orthogonality concept. Firstly, orthogonality is considered according to the model structure by expanding the Volterra kernels on an orthogonal basis of rational functions. The parsimony of the expansion is strongly linked to the basis selection. In order to get parsimonious expansions, we developed new tools for Laguerre bases and Generalized Orthonormal bases (GOB) optimization. Secondly, orthogonality is considered in relation with the input signals. By exploiting the statistical properties of the input signal, multivariable orthogonal polynomials were built. The parameters of the Volterra model expanded on such bases are then estimated without any matrix inversion, which significantly simplifies the parametric estimation in the MMSE sense. The input signals orthogonalization was also considered via a Gram-Schmidt procedure. In an adaptive context, the speed convergence of LMS type algorithms is accelerated without an excessive additional computational cost.

Some physical systems can be represented using a simplified Volterra model, with low parametric complexity, such as the Hammerstein and the Wiener models. That is the case of the Radio-over-fiber channel. We show particularly that a Wiener model and a Hammerstein model can respectively represent the corresponding uplink and downlink channels. In the SISO case, by using an input precoding, we suggest new algorithms for the joint estimation of the channel coefficients and the transmitted symbols in a semi-blind approach. For the uplink channel, a multi-sensors configuration is also

considered. For such a configuration, thanks to a specific input precoding, we exploit the spatial diversity introduced by the sensors and the temporal diversity to obtain a tensorial representation of the received signal. By applying the tensorial decomposition technique known as PARAFAC, we carry out the joint estimation of the channel and the transmitted symbols in a blind way.

Keywords : Modeling, Identification, Orthogonal Bases, Laguerre basis, Generalized orthonormal basis, orthogonal polynomials, pole optimization, parametric complexity reduction, equalization, Volterra model, Wiener model, Hammerstein model, PARAFAC decomposition.

Liste de Symboles

$*$	Convolution discrète
$(.)^*$	Opérateur de conjugaison d'un scalaire
$(.)^T$	Opérateur de transposition d'une matrice
$(.)^H$	Opérateur de transconjugaison d'une matrice
$(.)^\dagger$	Opérateur de pseudo-inversion matricielle
$\mathbb{E}$	Opérateur espérance mathématique
q	Opérateur avance
$\mathbf{\Gamma}$	Matrice d'autocorrélation
$\mathbf{C}$	Matrice d'intercorrélacion
$B_k(z)$	Fonction de base dans le domaine de la transformée en z
$b_k(n)$	Fonction de base dans le domaine temporel
$h_p(n_1, \dots, n_p)$	Noyau de Volterra d'ordre p
$u(n)$	Signal d'entrée
$s(n)$	Signal d'entrée filtré
$d(n)$	Signal désiré ou signal de référence
$y(n)$	Signal de sortie d'un modèle
μ	Pas d'adaptation
M	Mémoire d'un système
P	Ordre de non-linéarité
Π	Matrice de projection
Ω	Matrice de sélection de lignes
$C(M, P)$	Nombre de paramètres d'un modèle de Volterra de mémoire M et d'ordre P
n	Indice du temps ou nombre d'itérations
σ_x^2	Variance de la variable aléatoire x
δ_e	Retard d'égalisation
T	Période d'un signal périodique
$\otimes$	Produit de Kronecker
$\oslash$	Produit de Kronecker tronqué
$(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$	Représentation d'état
$\langle x, y \rangle$	Produit scalaire de x et de y
ξ	Pôle
$g_{k_1, \dots, k_p}$	Coefficient de Fourier

Liste d'abréviations

AIC	Critère d'information d'Akaike (<i>Akaike Information Criterion</i>)
ALS	Algorithme des moindres carrés alternés (<i>Alternating Least Squares</i>)
BFOR	Base de fonctions orthonormales rationnelles
BIC	Critère d'information Bayésien (<i>Bayesian Information Criterion</i>)
BOG	Base orthogonale généralisée
dB	Décibels
DD	Egaliseur dirigé par la décision (<i>Direct Decision equalizer</i>)
DFE	Egaliseur à retour de décision (<i>Decision Feedback Equalizer</i>)
EQM	Erreur Quadratique Moyenne
EQMEN	Erreur Quadratique Moyenne d'Estimation Normalisée
EQMM	Erreur Quadratique Moyenne Minimale
EQMN	Erreur Quadratique Moyenne Normalisée
HPA	Amplificateur de puissance (<i>High Power Amplifier</i>)
i.i.d	Distribué de manière identique et indépendante (<i>Independant and Identically Distributed</i>)
LMS	Algorithme des moindres carrés moyens (<i>Least Mean Squares</i>)
LSMN	Algorithme des moindres carrés moyens de type Newton
NLMS	Algorithme des moindres carrés moyens normalisé (<i>Normalized Least Mean Squares</i>)
OFDM	(<i>Orthogonal Frequency Division Multiplexing</i>)
PAM	Modulation à amplitude pulsée (<i>Pulse Amplitude Modulation</i>)
PARAFAC	<i>Parallel Factor Analysis</i>
PSK	Modulation à déplacement de phase (<i>Phase Shift Keying</i>)
QAM	Modulation à amplitude en quadrature (<i>Quadrature Amplitude Modulation</i>)
RIF	Réponse Impulsionnelle Finie
RII	Réponse Impulsionnelle Infinie
RLS	Algorithme des moindres carrés récursifs (<i>Recursive Least Squares</i>)
RF	Radio- fréquence
SIMO	Système mono-entrée multi-sorties (<i>Single Input Multiple Outputs</i>)
SISO	Système mono-entrée mono-sortie (<i>Single Input Single Output</i>)
SNR	Rapport signal à bruit (<i>Signal to Noise Ratio</i>)
SSPA	Amplificateur de puissance à semi-conducteurs (<i>Solid State Power Amplifier</i>)
SVD	Décomposition en valeurs singulières (<i>Singular Value Decomposition</i>)
VLSI	(<i>Very Large Scale Integration</i>)
TWTA	Amplificateur par tube à ondes progressives (<i>Travelling Wave Tube Amplifier</i>)

Table des matières

Liste des figures	xix
Liste des tableaux	xxiii
Introduction générale	1
1 Le modèle de Volterra	7
1.1 Introduction	7
1.2 Quelques généralités sur le modèle de Volterra	8
1.2.1 Ecriture vectorielle d'un filtre de Volterra	10
1.2.2 Ecriture matricielle d'un modèle homogène	13
1.2.3 La décomposition en coordonnées diagonales	14
1.3 Modèles de Volterra simplifiés	16
1.3.1 Modèles de Wiener et de Hammerstein	16
1.3.2 Modèle de Volterra quadratique à q diagonales	18
1.3.3 Modèle parallèle-cascade	19
1.4 Réduction de complexité par décomposition matricielle ou tensorielle .	19
1.4.1 Décomposition matricielle	19
1.4.2 Décomposition tensorielle	21
1.5 Estimation au sens de l'erreur quadratique moyenne minimale d'un modèle de Volterra	22
1.5.1 Condition d'inversibilité de la matrice d'autocorrélation du vec- teur de régression	23

1.5.2	Cas d'un modèle de Volterra du second ordre	23
1.5.3	Cas d'un modèle de Volterra cubique	24
1.6	Estimation adaptative	24
1.6.1	Algorithmes de type LMS	24
1.6.2	Algorithmes de type RLS	25
1.7	Exemples de canaux de communication non-linéaires	26
1.7.1	Communication par satellite	26
1.7.2	Systèmes de communication sans fil à large bande	27
1.7.3	Communication dans un environnement fermé	27
1.8	Egalisation des systèmes de Volterra	27
1.8.1	Inverse d'ordre P	28
1.8.2	Méthode du point fixe	28
1.8.3	Egalisation de type Wiener	29
1.8.4	Méthode des racines de polynômes	29
1.9	Conclusion	30
2	Les Bases de Fonctions Orthogonales Rationnelles	33
2.1	Introduction	33
2.2	Méthodes de construction des bases de fonctions orthogonales rationnelles	34
2.2.1	Preliminaires	35
2.2.2	Construction basée sur la représentation d'état équilibrée	37
2.2.3	Construction par orthogonalisation de Gram-Schmidt	41
2.2.4	Méthode générale de construction	47
2.3	Réseaux de filtres associés à la base de fonctions orthogonales rationnelles	49
2.3.1	Structure transversale	50
2.3.2	Structure orthogonale : Filtre en treillis	52
2.4	Choix de la base optimale	54
2.4.1	Position du problème	54

2.4.2	Choix des pôles	56
2.5	Choix de la meilleure base	62
2.5.1	Effet de l'ordre de placement des filtres passe-tout sur le choix de la base	62
2.5.2	Technique de seuillage	66
2.6	Conclusion	66
3	Réduction de complexité à l'aide de bases de fonctions orthogonales rationnelles	69
3.1	Introduction	69
3.2	Principe du développement des noyaux de Volterra sur une base orthogonale	70
3.3	Modèle de Volterra-Laguerre	72
3.3.1	Expression du pôle optimal en fonction du spectre de Laguerre .	74
3.3.2	Algorithme d'estimation en bloc du modèle de Volterra-Laguerre	77
3.3.3	Algorithme d'estimation adaptative	78
3.4	Modèle de Volterra-BFOR : Approche exhaustive	79
3.4.1	Cas du noyau de Volterra du premier ordre	79
3.4.2	Cas général	82
3.4.3	Modèle de Volterra-BFOR : Approche analytique	90
3.5	Approche hybride de réduction de complexité d'un modèle de Volterra .	94
3.6	Simulations	97
3.6.1	Cas linéaire	97
3.6.2	Cas non-linéaire	102
3.7	Conclusion	116
4	Méthodes orthogonales d'identification	119
4.1	Introduction	119
4.2	Bases de polynômes orthogonaux multivariables	120
4.2.1	Généralités sur les signaux complexes	121

4.2.2	Identification d'un modèle de Volterra cubique	122
4.2.3	Identification d'un modèle de Volterra passe-bande d'ordre trois à l'aide de signaux QAM et PSK	130
4.2.4	Estimation adaptative basée sur le formalisme des polynômes orthogonaux	136
4.2.5	Simulations	137
4.3	Structure orthogonale des filtres issus de la BFOR	142
4.3.1	Cas linéaire	143
4.3.2	Filtre de Volterra-BFOR adaptatif basé sur l'orthogonaliseur en escalier	151
4.3.3	Simulations	153
4.4	Estimation découplée pour une entrée gaussienne colorée	157
4.5	Conclusion	160
5	Contributions à l'identification et à l'égalisation aveugle et semi-aveugle	161
5.1	Introduction	161
5.2	Accès à un réseau de communication sans fil via la fibre optique	163
5.2.1	Système multi-capteurs	165
5.2.2	Modélisation	167
5.3	Identification et égalisation semi-aveugle de la liaison descendante	168
5.3.1	Transformation du modèle de Hammerstein en un modèle SIMO polynomial instantané	168
5.3.2	Identification d'un modèle SIMO polynomial statique	172
5.3.3	Simulation	176
5.3.4	Estimation des symboles transmis	179
5.4	Identification et égalisation de la liaison ascendante : cas mono-capteur	186
5.4.1	Simulation	188
5.5	Récepteur autodidacte de la liaison ascendante : cas multi-capteurs	189
5.5.1	Algorithme ALS pour l'estimation des paramètres	194
5.5.2	Simulations	195

5.6 Récepteur hybride TPS-root PARAFAC de la liaison ascendante	196
5.7 Conclusion	201
Conclusion générale et perspectives	203
Annexes	206
A Produit de Kronecker et décomposition PARAFAC	209
A.1 Le produit de Kronecker	209
A.2 La décomposition PARAFAC	210
B Preuves du chapitre 3	213
B.1 Preuve de la proposition 3.1	213
B.2 Preuve de la proposition 3.2	214
B.3 Preuve du lemme 3.1	214
B.4 Preuve de la proposition 3.3	216
C Polynômes orthogonaux	219
C.1 Polynômes orthogonaux pour des signaux PAM	219
C.1.1 Construction des polynômes orthogonaux monovariabiles pour une variable de type PAM	219
C.1.2 Liens avec les coefficients des noyaux du modèle de Volterra . .	220
C.2 Polynômes orthogonaux pour des signaux QAM	222
C.2.1 Construction des polynômes orthogonaux monovariabiles pour une variable de type QAM	222
C.2.2 Liens avec les coefficients des noyaux du modèle de Volterra . .	225
D Méthodes de Tseng et Powers	227
D.1 Cas d'un modèle cubique ayant une entrée <i>i.i.d.</i> symétrique	227
D.2 Cas d'un modèle cubique passe-bande ayant une entrée <i>i.i.d.</i> de type QAM	228
Bibliographie	229

Liste des figures

1.1	Implémentation multicanal d'un modèle de Volterra homogène d'ordre p	16
1.2	Modèle de Wiener	17
1.3	Modèle de Hammerstein	18
1.4	Modèle parallèle-cascade	19
2.1	Mise en cascade de deux passe-tout	48
2.2	Réseau de filtres de la BFOR de Ninness et Gustafsson	50
2.3	Réseau de filtres de la BFOR de Ninness et Gustafsson : cas des pôles réels	52
2.4	Modèle de Hambo ou réseau de filtres de la BFOR de Heuberger <i>et al.</i> .	53
2.5	Borne supérieure de l'erreur de modélisation pour différents choix de pôles	57
2.6	Cascade de filtres générant les fonctions de base $B_{k;l}$ et $B_{k+1;l}$	64
2.7	Cascade de filtres après interversion des pôles	64
3.1	Modèle de Volterra représenté à l'aide d'un banc de filtres orthogonaux	71
3.2	Représentation multicanal équivalente du modèle de Volterra-BFOR orthogonalisé	93
3.3	Représentation d'un modèle de Volterra BFOR-SVD quadratique homogène à noyau séparable	96
3.4	Lieu des pôles et des zéros du système simulé	98
3.5	Erreur quadratique normalisée par le nombre de données en fonction de l'ordre de troncature	99
3.6	Coefficients de Fourier de la base sélectionnée	99

3.7	Comparaison des développements sur les bases RIF, Laguerre et BFOR	100
3.8	Coefficients de Fourier de la meilleure base au sens de l'entropie minimale	101
3.9	Coefficients de Fourier de la plus mauvaise base au sens de l'entropie minimale	102
3.10	Courbes théoriques des fonctions de coût J_1 (trait plein) et J_2 (trait discontinu)	104
3.11	Estimation des pôles	105
3.12	Evolution de l'EQMN pour différents ordres de troncature	106
3.13	Estimation des pôles pour différentes valeurs initiales des pôles de Laguerre	106
3.14	Estimation des pôles (à droite) et évolution de $R_{2,p}$ (à gauche)	107
3.15	Estimation des pôles de Laguerre (en haut) et de quelques coefficients de Fourier coefficients (en bas) associés au noyau du premier ordre (à gauche) et du second ordre (à droite)	108
3.16	Comparaison de trois configurations du filtre de Volterra-Laguerre : (1) Pôles de Laguerre arbitrairement fixés (2) Pôles de Laguerre estimés avec l'algorithme proposé (3) Pôles de Laguerre égaux aux pôles optimaux	108
3.17	EQMN en fonction de la dimension de la base pour un développement sur deux bases	110
3.18	Comparaison de la sortie du système avec sa reconstruction obtenue à l'aide du modèle de Volterra-BFOR optimal (SNR=30 dB)	111
3.19	Comparaison des développements sur une base unique et sur des bases distinctes	111
3.20	Comparaison des représentations sur des bases de Laguerre et des bases BFOR	112
3.21	EQMN en fonction de l'ordre de troncature du développement du noyau du premier ordre	113
3.22	EQMN en fonction de l'ordre de troncature du développement du noyau du second ordre	114
3.23	Puissance du signal résiduel en fonction de la dimension de la base . . .	115
3.24	Comparaison des différentes approches	116

4.1	Estimation adaptative basée sur le formalisme des polynômes orthogonaux	137
4.2	EQMN en fonction du rapport signal à bruit pour un modèle cubique sans mémoire	138
4.3	EQMN en fonction du rapport signal à bruit	139
4.4	Erreur quadratique d'estimation d'un modèle cubique : (1) LMS, (2) LMS basé sur le formalisme des polynômes orthogonaux	140
4.5	EQMN en fonction du rapport signal à bruit pour une entrée de type QAM-16	141
4.6	Structure en escalier pour l'orthogonalisation des entrées filtrées par la BFOR pour $K = 4$	145
4.7	Filtrage adaptatif de Volterra avec structure orthogonale	152
4.8	Efficacité d'orthogonalisation	154
4.9	Variation des coefficients de l'orthogonaliseur	155
4.10	Comparaison de trois algorithmes adaptatifs pour une entrée gaussienne colorée (BFOR tronquée à $K = 3$)	156
4.11	Comparaison de trois algorithmes d'adaptation d'un filtre de Volterra	157
4.12	Estimation d'un modèle de Volterra à complexité réduite à l'aide des polynômes orthogonaux	158
4.13	Nombre condition de la matrice des moments de l'entrée en fonction de la coloration du signal	160
5.1	Architecture d'un réseau sans fil avec utilisation de la fibre optique	163
5.2	Schéma fonctionnel d'une liaison ascendante	164
5.3	Schéma fonctionnel d'une liaison descendante	165
5.4	Configuration ayant plusieurs liaisons optiques	166
5.5	Configuration ayant une liaison optique partagée	166
5.6	Cas d'un modèle de Hammerstein d'ordre 3, ($M = 2$)	173
5.7	Evaluation de la précision de l'estimation pour différentes périodes de la séquence pilote	179
5.8	Taux d'erreur d'estimation des symboles de l'algorithme TPS-root pour différentes périodes de la séquence pilote	184

5.9	Taux d'erreur d'estimation des symboles de l'algorithme PI pour différentes périodes de la séquence pilote	185
5.10	Comparaison des performances des algorithmes TPS-root, PI et de la méthode du point fixe	185
5.11	Taux d'erreur d'estimation des symboles en fonction du niveau de bruit (N=500)	196
A.1	Décomposition PARAFAC	210
A.2	Matrices horizontales, verticales et latérales $\mathbf{H}_{n_1..}$, $\mathbf{H}_{..n_2}$, et $\mathbf{H}_{..n_3}$	211

Liste des tableaux

2.1	Bases générées par la mise en cascade de l'ensemble de passe-tout $\{G_i\}_{i=0}^2$	63
3.1	Pôles sélectionnés	98
3.2	Valeur de l'entropie pour différentes configurations de l'emplacement des pôles	101
3.3	Valeurs optimales des pôles en fonction de l'ordre de troncature	104
3.4	Ordre de troncature optimal d'une base unique	109
3.5	Ordre de troncature optimal des deux bases	110
3.6	Complexité paramétrique	113
3.7	Technique de seuillage ($N = 200$ et $SNR = 30\text{dB}$)	114
4.1	Base orthonormale de $\mathbb{C}_3[x_1, x_2, x_3]$	127
4.2	Comparaison avec la méthode EQM classique dans le cas d'un modèle cubique sans mémoire ($N = 12000$)	138
4.3	EQMN pour une entrée PAM-4 ($N = 16000$)	139
4.4	Conditionnement de la matrice des moments de l'entrée d'un modèle de Volterra cubique pour une entrée PAM-4	139
4.5	Coefficients du modèle de Volterra d'ordre trois représentant un canal satellitaire	141
4.6	EQMN pour une entrée QAM-16 ($N = 16000$)	142
4.7	Conditionnement de la matrice des moments de l'entrée d'un modèle de Volterra cubique ; entrée de type QAM-16 et QAM-64	142
4.8	Nombre d'opérations élémentaires à chaque itération	151
4.9	Nombre condition de la matrice d'autocorrélation de l'entrée	155

5.1	Estimation paramétrique en utilisant une séquence PAM-4	177
5.2	Estimation paramétrique en utilisant une séquence gaussienne	177
5.3	Estimation paramétrique en utilisant une séquence pilote périodique . .	178
5.4	Evaluation de la mesure EQMEN pour différentes séquences pilotes . .	178
5.5	Estimation des coefficients du noyau d'ordre 1 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$	189
5.6	Estimation des coefficients du noyau d'ordre 2 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$	189
5.7	Estimation des coefficients du noyau d'ordre 3 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$	190
5.8	Evaluation de l'EQMEN pour différentes périodes de la séquence pilote	190

Introduction générale

En traitement du signal, en automatique et, d'une manière générale, dans toutes les sciences appliquées, il est souvent nécessaire de décrire le comportement d'un système par un modèle mathématique. En effet, la donnée d'un modèle mathématique peut servir en simulation (dans le but de comprendre et de prédire le comportement d'un système, de remplacer une expérimentation sur site, ...), pour la synthèse d'un correcteur pour la commande des systèmes, pour la classification des signaux observés, pour la synthèse d'un égaliseur ou d'un annulateur d'échos ou de bruit, etc. La *Modélisation*, art d'élaborer un modèle, se trouve donc au cœur de plusieurs disciplines scientifiques. Elle peut être menée au moyen d'une analyse détaillée du phénomène décrit par le système tout en appliquant les différentes lois physico-chimiques, biologiques, économiques ou autres régissant ce système. De cette approche résulte un modèle dit *modèle de connaissance*. Elle peut aussi être menée à partir d'une analyse expérimentale des données d'entrée/sortie dont le but est de reproduire le comportement externe du système. Il en résulte un *modèle de représentation* ou *modèle boîte noire*.

L'estimation paramétrique d'un modèle est essentiellement fonction de cinq facteurs [34] :

- la structure du modèle à identifier ;
- l'information *a priori* disponible sur le système et son environnement ;
- les signaux utilisés pour réaliser l'identification ;
- le critère à optimiser ;
- l'algorithme d'optimisation.

Les structures de modèle peuvent être classées en plusieurs catégories suivant que le système à identifier est monovariable ou multivariable, déterministe ou stochastique, à temps continu ou discret, linéaire ou non, invariant dans le temps ou non. Dans ce mémoire, la structure de modèle considérée est en général monovariable¹, à temps discret, invariant dans le temps et non-linéaire.

Depuis quelques années, la modélisation non-linéaire des systèmes est l'objet d'une attention particulière de la part de plusieurs équipes de chercheurs (voir les références citées dans [47]). Cet intérêt vient du fait que de nombreux systèmes rencontrés dans la nature ou dans différents domaines de l'ingénierie sont non-linéaires. Dans le domaine

¹Le cas particulier d'un modèle multivariable est aussi traité mais dans le cas d'un système de communication précis

des télécommunications, par exemple, il a été montré que les systèmes de transmission opérant à des taux supérieurs à 4800 bits/s sont soumis à des distorsions non linéaires [80], que les amplificateurs de puissance sont utilisés à la limite de la saturation et présentent un comportement fortement non-linéaire, qu'en transmission sur fibre optique, la conversion d'un signal électrique en un signal optique est source de distorsion non-linéaire, etc.

Il existe une grande variété de modèles non-linéaires. Une première tâche est donc de déterminer la structure la plus appropriée au système à représenter. Une bonne revue des structures non-linéaires peut être obtenue dans [12, 121]. Parmi ces structures, les modèles polynômiaux figurent parmi les plus utilisés. Dans ce mémoire, la structure de modèle considérée est polynômiale de type Volterra.

Le modèle de Volterra, bien qu'étant non-linéaire par rapport à l'entrée, est linéaire par rapport à ses paramètres. Cet atout permet de lui appliquer un grand nombre de résultats et de techniques d'identification développés dans le cas linéaire. En outre, il peut être vu comme une extension naturelle de la notion de réponse impulsionnelle des systèmes linéaires aux systèmes non-linéaires. Cependant, son principal écueil est l'accroissement très rapide du nombre de ses paramètres en fonction de la mémoire et de l'ordre de non-linéarité du système à représenter. L'estimation d'un nombre trop élevé de paramètres peut en effet être problématique. En filtrage adaptatif, par exemple, la complexité de calcul des algorithmes est liée au nombre de paramètres à estimer. En conséquence, dans la littérature, de nombreux auteurs ne traitent que des cas de modèle de Volterra de mémoire et d'ordre très modestes. Outre cet aléa structurel, l'estimation paramétrique d'un modèle de Volterra, au sens de l'erreur quadratique moyenne minimale, revêt une certaine complexité liée au conditionnement des matrices à traiter. Ceci rejaillit aussi sur la rapidité de convergence des algorithmes adaptatifs de type LMS. Nous consacrons donc la majeure partie de cette thèse à ces deux questions : complexité paramétrique et complexité des procédures d'identification au sens de l'erreur quadratique moyenne minimale. S'il est un maître mot qui puisse caractériser les solutions apportées, c'est bien celui de l'*orthogonalité*. Orthogonalité vis-à-vis de la structure, en développant les noyaux de Volterra sur une base orthogonale, en vue d'obtenir une représentation parcimonieuse du système à identifier ; et orthogonalité vis-à-vis des signaux d'entrée basée sur les propriétés statistiques de ces derniers. Les outils développés dans cette thèse en matière de réduction de complexité paramétrique ou d'estimation paramétrique ont un spectre d'application large. Entre autres, ils peuvent notamment servir dans le domaine de la reconstruction des signaux transmis par un système de communication, plus précisément pour l'identification d'un canal de transmission non-linéaire, en vue de synthétiser un égaliseur.

Certains systèmes physiques peuvent être représentés à l'aide d'un modèle de Volterra simplifié, à faible complexité paramétrique, tel que le modèle de Hammerstein et celui de Wiener. C'est le cas d'un canal de communication représentant l'accès à un réseau sans fil via une fibre optique. Les liaisons montante et descendante de ce canal peuvent être représentées respectivement par un modèle de Wiener et par un modèle de Hammerstein. L'identification et/ou l'égalisation aveugles ou semi-aveugles de ce type de

canal ont été très peu traitées dans la littérature. Vu l'intérêt de ce système de communication, nous développons de nouvelles méthodes déterministes d'identification et d'égalisation semi-aveugles et aveugles.

La présente thèse s'articule autour des trois axes suivants :

- La réduction de la complexité paramétrique d'un modèle de Volterra ;
- La réduction de la complexité des méthodes d'estimation paramétrique d'un modèle de Volterra, au sens de l'erreur quadratique moyenne minimale ;
- L'identification et l'égalisation semi-aveugles et aveugles de systèmes représentés par un modèle de Volterra simplifié de type Hammerstein ou Wiener.

Le traitement de ces trois axes a été effectué sur cinq chapitres dont l'organisation est la suivante :

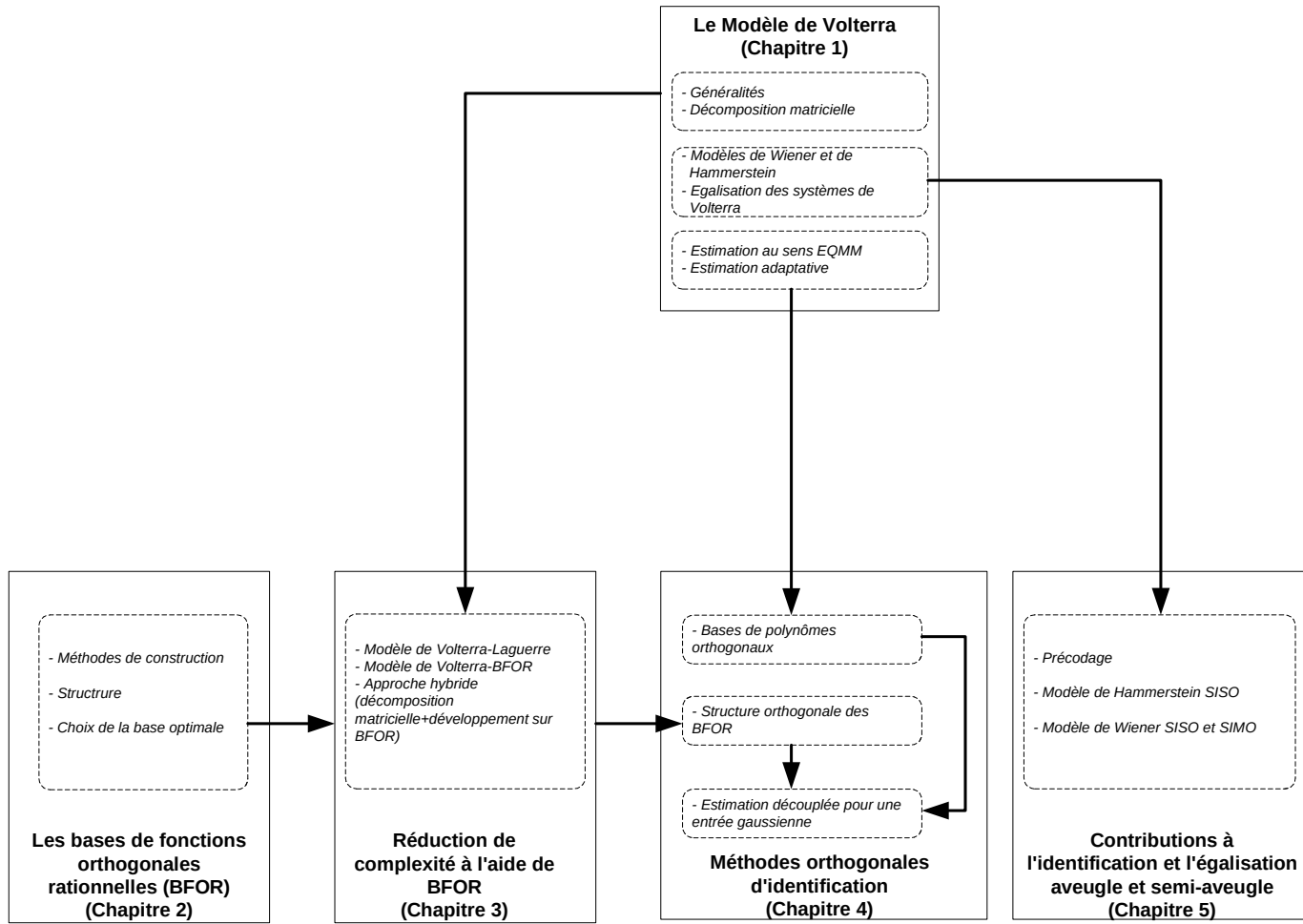
Les chapitres 1 et 2 sont respectivement une revue bibliographique sur le modèle de Volterra et sur les bases de fonctions orthogonales. Dans le premier chapitre nous mettons en évidence les deux problèmes clés liés à l'identification des modèles de Volterra comme précédemment indiqué. Avant de proposer des solutions à ces problématiques, dans le second chapitre, nous donnons une présentation unifiée des bases de fonctions orthogonales rationnelles, outil principal de l'approche de réduction de complexité que nous considérons.

Dans le chapitre 3, nous apportons des solutions à la réduction de complexité paramétrique du modèle de Volterra à l'aide des bases de fonctions orthogonales présentées dans le chapitre 2. Trois types de solutions étayées par des simulations numériques sont proposées.

Dans le chapitre 4, la question de la réduction de la complexité de l'estimation paramétrique du modèle de Volterra est abordée suivant deux axes : la représentation du modèle à estimer sur des bases de polynômes orthogonaux construits à partir des propriétés statistiques des signaux d'entrée et l'orthogonalisation des signaux d'entrée via un orthogonaliseur de Gram-Schmidt.

Enfin, le chapitre 5 traite des méthodes d'identification et d'égalisation semi-aveugles et aveugles dans le cas d'un système de communication précis : l'accès à un réseau sans fil par le biais d'une fibre optique. Ce système est décrit puis modélisé à l'aide du modèle de Wiener pour la liaison montante et du modèle de Hammerstein pour la liaison descendante.

Les liens entre les différents chapitres sont indiqués par le diagramme de la page suivante.



Les principales contributions de cette thèse sont les suivantes :

- Définition du produit de Kronecker tronqué (section 1.2.1.1) ;
- Unification des méthodes de construction des bases de fonctions orthogonales rationnelles au travers d'une synthèse bibliographique (section 2.2) ;
- Développement de méthodes d'optimisation des bases de Laguerre (section 3.3) ;
- Développement de méthodes d'optimisation des bases de fonctions orthogonales rationnelles selon la construction de Ninness et Gustafsson [90] (sections 3.4 et 3.5) ;
- Construction de bases de polynômes orthogonaux multivariables pour des signaux de type PAM et QAM (section 4.2) ;
- Détermination d'un module d'orthogonalisation de signaux en sortie des filtres issus des bases de fonctions orthogonales rationnelles de Ninness et Gustafsson (section 4.3) ;
- Développement de techniques d'estimation semi-aveugle basées sur un précodage des entrées (sections 5.3 et 5.4) ;
- Enoncé d'un théorème permettant de déterminer les coefficients non diagonaux d'un modèle de Wiener à partir de la connaissance des coefficients diagonaux (section 5.4) ;
- Modélisation d'un canal de transmission non linéaire sous forme tensorielle (section 5.5) ;
- Développement d'un récepteur aveugle basé sur une décomposition tensorielle de type PARAFAC dans le cas des systèmes multi-capteurs (sections 5.5 et 5.6).

Certaines de ces contributions ont donné lieu aux publications suivantes :

- Articles soumis à des revues internationales
 1. Kibangou A.Y., G. Favier and M.M. Hassani, "Selection of Generalized Orthonormal Bases for Second-Order Volterra Filters", à paraître dans *Signal Processing*.
 2. Kibangou A.Y., G. Favier and M.M. Hassani, "Optimization of Laguerre-Volterra filters based on Laguerre Spectra", soumis à *EURASIP Journal on Applied Signal Processing* (2004)
- Communications parues dans des actes de conférences internationales
 1. Kibangou A.Y., G. Favier and M.M. Hassani, "Algorithme Hiérarchique pour l'estimation optimale des noyaux de Volterra développés sur la base de Laguerre", In *Actes du 6è Colloque Africain sur la Recherche en Informatique (CARI'02)*, pp. 395-402, Yaoundé, Cameroun, Octobre 2002.
 2. Kibangou A.Y., G. Favier, M.M. Hassani and A. Zeroual, "Estimation optimale des noyaux de Volterra développés en séries de Laguerre", In *proc. of Conférence internationale de mathématiques appliquées aux sciences de l'ingénieur (CIMASI)*, Casablanca, Maroc, Octobre 2002.
 3. Kibangou A.Y., G. Favier and M.M. Hassani, "A growing approach for selecting generalized orthogonal basis functions in the context of system modelling", *Proc. IEEE-EURASIP workshop on Nonlinear Signal and Image processing (NSIP)*, Grado-Trieste, Italy, June 2003.

4. Kibangou A.Y., G. Favier and M.M. Hassani, "Generalized orthonormal basis selection for expanding quadratic Volterra filters", *Proc. IFAC Symposium on System Identification (SYSID)*", pp. 1119-1124, Rotterdam, The Netherlands, August 2003.
 5. Favier G., A.Y. Kibangou and R.G.J.B. Campello, "Nonlinear system modelling by means of orthonormal basis functions" , *invited paper at 2nd IEEE Conference on Signal, Systems, Decision and information theory (SSD)*, Sousse, Tunisia, March 2003.
 6. Khouaja A., A.Y. Kibangou and G. Favier, "Third-order Volterra kernels complexity reduction using PARAFAC", In *Proc. IEEE International Symposium on Control, Communications and Signal Processing (ISCCSP)*, pp. 857-860, Hammamet, Tunisia, March 2004.
 7. Favier G., A.Y. Kibangou and A. Khouaja, "Nonlinear system modelling by means of Volterra models. Approaches for the parametric complexity reduction", *Invited paper in Symposium Techniques Avancées et Stratégies Innovantes en Modélisation et commandes robustes des processus industriels*, Martigues, France, September, 2004.
- Communications soumises et acceptées à des conférences internationales
 1. Kibangou A.Y., G. Favier and M.M. Hassani, "Iterative optimization method of GOB-Volterra filters", accepté pour publication à *IFAC World Congress 2005*
 - Communications soumises à des conférences internationales
 1. Kibangou A.Y., G. Favier and M.M. Hassani, "Adaptive Laguerre-Volterra filters optimization based on Laguerre spectra", soumis à *EUropean Signal and Image Processing Conference (EUSIPCO 2005)*

Chapitre 1

Le modèle de Volterra

1.1 Introduction

Les séries de Volterra, dues au mathématicien italien du même nom, ont été présentées pour la première fois en 1887. Cependant, leur utilisation pour l'analyse des systèmes non-linéaires a été, pour la première fois, effectuée par N. Wiener dans le but de caractériser la réponse d'un système non-linéaire. Après ce premier travail, Wiener s'est consacré à différents problèmes portant sur l'analyse et la synthèse de systèmes non-linéaires. Plusieurs autres chercheurs ont poursuivi cet axe d'étude (voir [121, 12] et les références qui y sont citées). L'extension de l'utilisation des séries de Volterra aux systèmes non-linéaires discrets a été réalisée pour la première fois par Alper en 1965 [5].

Le modèle de Volterra appartient à la classe des modèles polynômiaux non récurrents. La relation d'entrée-sortie qui lui est associée est :

$$y(n) = \sum_{p=1}^{\infty} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j) \quad (1.1)$$

où u , y et h_p sont respectivement l'entrée, la sortie et le noyau de Volterra d'ordre p ; les n_j , $j = 1, \dots, p$ étant des entiers. Un noyau d'ordre p peut être vu comme une généralisation à l'ordre p de la notion de réponse impulsionnelle bien connue dans le cas des systèmes linéaires. Boyd et Chua [18] ont montré que tout système non-linéaire causal, invariant dans le temps et de mémoire finie peut être représenté par un développement fini en série de Volterra. En appelant M la mémoire du système, on peut donc considérer le modèle tronqué à l'ordre P suivant :

$$y(n) = \sum_{p=1}^P \sum_{n_1=0}^{M-1} \sum_{n_2=0}^{M-1} \cdots \sum_{n_p=0}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j) \quad (1.2)$$

Comme on peut le constater, avec les relations ci-dessus, le modèle de Volterra est un modèle non linéaire qui présente l'avantage d'être linéaire par rapport à ses paramètres, la non linéarité s'exprimant par rapport à l'entrée du système. L'implication pratique de cette propriété est que plusieurs des résultats obtenus pour des problèmes d'estimation paramétrique linéaire peuvent être repris pour ce modèle, en prenant toutefois le soin d'apporter quelques aménagements qui s'imposent.

Il est à signaler que la représentation paramétrique du modèle de Volterra nécessite un nombre de paramètres élevé. C'est là, l'un des principaux défauts qui limitent l'utilisation de ce modèle dans des applications pratiques. Afin de surmonter ce défaut, deux axes peuvent être considérés. Le premier consiste à considérer des modèles de Volterra simplifiés comme le modèle de Wiener ou de Hammerstein, i.e. la mise en cascade d'un filtre linéaire avec une non-linéarité statique. Le second consiste à utiliser des méthodes de réduction de complexité. Celles-ci peuvent être classées en deux catégories : les décompositions matricielles ou tensorielles des matrices ou des tenseurs constitués par les coefficients des noyaux et les développements sur des bases de fonctions orthogonales que nous allons présenter dans le chapitre 2. Nous nous focaliserons sur cette approche de réduction de complexité dans le chapitre 3.

Cependant, même dans le cas d'un modèle de Volterra ayant un nombre de paramètres raisonnable, l'estimation paramétrique au sens de l'erreur quadratique moyenne est confrontée à la manipulation de matrices denses souvent mal conditionnées. Il est donc nécessaire d'améliorer le conditionnement des dites matrices et de formuler des méthodes d'estimation ayant une complexité de calcul faible.

Dans le domaine des communications, certains composants présentent un comportement non-linéaire. Représenter les canaux de communications utilisant ce type de composants par un modèle de Volterra est donc une approche toute naturelle que plusieurs auteurs ont considérée dans la littérature. Nous considérerons donc la question de l'égalisation de tels canaux.

1.2 Quelques généralités sur le modèle de Volterra

Le modèle de Volterra (1.1) est causal. Il est dit strictement causal si ses noyaux sont tels que :

$$h_p(n_1, \dots, n_p) = 0 \quad \text{si} \quad \exists i \in \{1, \dots, p\} : n_i = 0 \quad (1.3)$$

Il est dit stable au sens entrée bornée- sortie bornée (*BIBO : Bounded Input Bounded Output*) si et seulement si ses noyaux sont absolument sommables :

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} |h_p(n_1, \dots, n_p)| < \infty \quad (1.4)$$

Si toutes les permutations des indices $n_1, \dots, n_p$ laissent les coefficients du noyau $h_p(n_1, \dots, n_p)$ inchangés, ce noyau est dit symétrique.

Par ailleurs, il est à noter que la relation (1.2) peut être réécrite comme suit :

$$y(n) = \sum_{p=1}^P \bar{h}_p[u(n)]$$

avec

$$\bar{h}_p[u(n)] = \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=0}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j)$$

Ce terme est le terme homogène d'ordre p . On désigne donc par modèle de Volterra homogène d'ordre p un modèle dont le seul terme homogène non nul est $\bar{h}_p$. Comme annoncé en introduction, l'inconvénient majeur est l'augmentation de la complexité paramétrique du modèle de manière exponentielle avec l'ordre du système. En effet, le nombre de coefficients contenus dans un modèle d'ordre P et de mémoire M est :

$$C(M, P) = \sum_{p=1}^P M^p \quad (1.5)$$

Toutefois, on peut constater que certains coefficients des noyaux pondèrent les mêmes produits des entrées $u(n)$. Chaque permutation $\pi(\cdot)$ des indices $n_1, \dots, n_p$ ne modifie pas le produit d'ordre p suivant $u(n - n_1) \cdots u(n - n_p)$. Ce qui signifie qu'on a :

$$\prod_{j=1}^p u(n - n_j) = \prod_{j=1}^p u(n - n_{\pi(j)})$$

En regroupant ces termes, on obtient une forme symétrique ou une forme triangulaire non redondante du noyau. En notant $\text{Card}(\pi(n_1, \dots, n_p))$ le nombre total de permutations des indices, la forme symétrique¹ est donnée par :

$$h_p^{\text{sym}}(n_1, \dots, n_p) = \frac{\sum_{\pi(\cdot)} h_p(n_{\pi(1)}, \dots, n_{\pi(p)})}{\text{Card}(\pi(n_1, \dots, n_p))} \quad (1.6)$$

Par suite, le modèle de Volterra triangulaire est :

$$y(n) = \sum_{p=1}^P \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \cdots \sum_{n_p=n_{p-1}}^M h_p^{\text{tri}}(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j) \quad (1.7)$$

où

$$h_p^{\text{tri}}(n_1, \dots, n_p) = \begin{cases} \sum_{\pi(\cdot)} h_p(n_{\pi(1)}, \dots, n_{\pi(p)}), & 0 \leq n_1 \leq \dots \leq n_p \leq M-1 \\ 0, & \text{sinon} \end{cases} \quad (1.8)$$

La réduction du nombre de paramètres est très significative puisque chaque noyau d'ordre p n'est plus caractérisé que par $C_{M,p}$ paramètres, avec

$$C_{M,p} = \frac{(M+p-1)!}{(M-1)!p!}$$

¹Dans la littérature, on fait souvent l'hypothèse de symétrie des noyaux. Cette hypothèse est non restrictive vu qu'il est toujours possible de rendre symétrique un noyau quelconque.

Par suite, le modèle complet ne contient plus que

$$C^{tri}(M, P) = \sum_{p=1}^P C_{M,p} \quad (1.9)$$

Pour un modèle homogène d'ordre p la réduction de complexité est mesurée par le ratio suivant :

$$\frac{C^{tri}(M, p)}{C(M, p)} = \frac{(M + p - 1)!}{(M - 1)! p! M^p} \quad (1.10)$$

Exemple 1.1 Pour un modèle homogène d'ordre $p = 4$, de mémoire 5 nous avons une réduction de complexité de $\frac{126}{1296}$ soit environ $1/10$.

1.2.1 Ecriture vectorielle d'un filtre de Volterra

La représentation triangulaire entrée-sortie (1.7) admet l'écriture vectorielle suivante :

$$y(n) = \Theta^T U(n) \quad (1.11)$$

où Θ est le vecteur contenant les coefficients des noyaux et $U(n)$ celui contenant les produits du signal d'entrée nécessaires à l'évaluation de la sortie $y(n)$.

$$\begin{aligned} U_1(n) &= [u(n) \ u(n-1) \cdots u(n-M+1)]^T \\ U(n) &= [U_1^T(n) \cdots U_p^T(n)]^T \\ \Theta &= [\Theta_1^T(n) \cdots \Theta_p^T(n)]^T \end{aligned}$$

Chaque vecteur non redondant $U_p(n)$, contenant les produits du signal nécessaires au terme homogène d'ordre p , est obtenu à partir du vecteur de données correspondant à la partie linéaire U_1 en faisant :

$$U_p(n) = \mathbf{\Omega} (\otimes_1^p U_1(n))$$

où $\otimes$ représente le produit de Kronecker dont la définition est rappelée en annexe A.1, $\otimes_1^p U_1(n) = \underbrace{U_1(n) \otimes \cdots \otimes U_1(n)}_{p \text{ termes}}$ et $\mathbf{\Omega}$ est une matrice de sélection de lignes de dimension

$\frac{(M-p+1)!}{(M-1)! p!} \times M^p$. On peut dire que $U_p(n)$ est le *produit de Kronecker tronqué* d'ordre p de $U_1(n)$.

1.2.1.1 Notion de produit de Kronecker tronqué

Définition 1.1 Etant donné le vecteur $X = [x_1, \dots, x_n]$ de dimension n , on désigne par produit de Kronecker tronqué de X par lui-même, noté $X \oslash X$, le sous vecteur de $X \otimes X$ de dimension $\frac{n(n+1)}{2}$, tel que :

$$X \oslash X = \mathbf{\Omega}(X \otimes X)$$

où Ω est une matrice de sélection de lignes formée par les vecteurs lignes e_k^T où e_k représente le k -ième vecteur de la base canonique de l'espace $\mathbb{R}^{n^2}$, k étant pris tel que $k = (j-1)n + i$; $j = 1, \dots, n$; $i = 1, \dots, n$.

Considérons le vecteur $X_p = [c_1 \cdots c_{k_p} \cdots c_{n^p}]^T$ obtenu par $(p-1)$ produits de Kronecker du vecteur X par lui-même.

Proposition 1.1 La position de l'élément $c_{k_p(i_1, \dots, i_p)} = \prod_{j=1}^p x_{i_j}$ dans le vecteur X_p est donnée par :

$$k_p(i_1, \dots, i_p) = \sum_{j=1}^{p-1} (i_j - 1)n^{p-j} + i_p$$

avec $i_j = 1, \dots, n$; $j = 1, \dots, p$

Preuve : Considérons un élément de X_2 situé à la position k_2 qui soit le produit des éléments de X , respectivement situés aux positions i_1 et i_2 . Tenant compte de la définition du produit de Kronecker de X par lui-même, cette position est donnée par : $k_2(i_1, i_2) = (i_1 - 1)n + i_2$; $i_1 = 1, \dots, n$ et $i_2 = 1, \dots, n$.

Supposons que la position de $c_{k_{p-1}}$ dans le vecteur X_{p-1} soit donnée par :

$$k_{p-1}(i_1, \dots, i_{p-1}) = \sum_{j=1}^{p-2} (i_j - 1)n^{p-1-j} + i_{p-1}, \quad i_j = 1, \dots, n; j = 1, \dots, p-1$$

Déterminons alors la position de c_{k_p} dans le vecteur X_p .

Sachant que

$$X_p = X_{p-1} \otimes X$$

l'élément $c_{k_p(i_1, \dots, i_p)}$ est le produit de $c_{k_{p-1}(i_1, \dots, i_{p-1})}$, élément de X_{p-1} , et de x_{i_p} , élément de X . Conformément à la définition rappelée en annexe A.1, sa position est donnée par :

$$k_p(i_1, \dots, i_p) = (k_{p-1}(i_1, \dots, i_{p-1}) - 1)n + i_p$$

avec

$$1 \leq k_{p-1}(i_1, \dots, i_{p-1}) \leq n^{p-1}; i_p = 1, \dots, n$$

Tenant compte de l'hypothèse de récurrence, alors :

$$\begin{aligned}
 k_p(i_1, \dots, i_p) &= \left(\sum_{j=1}^{p-2} (i_j - 1)n^{p-1-j} + i_{p-1} - 1 \right) n + i_p \\
 &= \sum_{j=1}^{p-2} (i_j - 1)n^{p-j} + (i_{p-1} - 1)n + i_p \\
 &= \sum_{j=1}^{p-1} (i_j - 1)n^{p-j} + i_p
 \end{aligned}$$

avec $i_j = 1, \dots, n$; $j = 1, \dots, p$. Ce qui est le résultat recherché. ■

On sait donc déterminer la position de chaque élément de X_p . La sélection non redondante qui génère le sous-vecteur de X_p correspondant au produit de Kronecker tronqué de X par lui-même est alors effectuée en prenant les indices i_l de la manière suivante $i_l = i_{l-1}, \dots, n$; $l = 1, \dots, p$ et $i_0 = 1$.

Définition 1.2 Le vecteur X_{pt} , obtenu par p produits de Kronecker tronqués de X par lui-même, est défini par :

$$X_{pt} = \mathbf{\Omega} X_p$$

où $\mathbf{\Omega}$ est une matrice de sélection de lignes formée à partir des vecteurs lignes $e_{k_p(i_1, \dots, i_p)}^T$, $e_{k_p(i_1, \dots, i_p)}$ étant le $k_p(i_1, \dots, i_p)$ -ième vecteur de la base canonique de l'espace $\mathbb{R}^{n^p}$, avec :

$$\begin{aligned}
 k_p(i_1, \dots, i_p) &= \sum_{j=1}^{p-1} (i_j - 1)n^{p-j} + i_p \\
 i_l &= i_{l-1}, \dots, n; \quad l = 1, \dots, p; \quad i_0 = 1
 \end{aligned}$$

Les éléments $c_{k_p(i_1, \dots, i_p)}$ de X_{pt} sont tels que :

$$c_{k_p(i_1, \dots, i_p)} = \prod_{j=1}^p x_{i_j}$$

Les valeurs possibles des indices k_p et i_j sont données par la définition 1.2. Le vecteur X_{pt} est un sous vecteur de X_p de dimension $\frac{(n+p-1)!}{p!(n-1)!}$.

Exemple 1.2 Considérons un noyau cubique de mémoire $M = 2$. On a donc

$$U_1(n) = (u(n) u(n-1))^T$$

D'où :

$$\begin{aligned}
 U_1(n) \otimes U_1(n) \otimes U_1(n) &= \begin{pmatrix} u^3(n) & u^2(n)u(n-1) & u^2(n)u(n-1) & u(n)u^2(n-1) \\ u^2(n)u(n-1) & u(n)u^2(n-1) & u(n)u^2(n-1) & u^3(n-1) \end{pmatrix}^T
 \end{aligned}$$

Pour obtenir le vecteur non redondant nous appliquons la matrice de sélection de lignes $\mathbf{\Omega}$. En considérant la définition 1.2, $\mathbf{\Omega}$ est formée par les vecteurs $e_{k_3(i_1, i_2, i_3)}$ avec $i_j = i_{j-1}, \dots, 2$, $j = 1, 2, 3$ et $i_0 = 1$. Les valeurs admises par $k_3(i_1, i_2, i_3)$ sont telles que :

$$k_3(i_1, i_2, i_3) = (i_1 - 1)M^2 + (i_2 - 1)M + i_3$$

i.e.

$$\begin{aligned} k_3(1, 1, 1) &= 1 \\ k_3(1, 1, 2) &= 2 \\ k_3(1, 2, 2) &= 4 \\ k_3(2, 2, 2) &= 8 \end{aligned}$$

Par suite :

$$\mathbf{\Omega} = \begin{pmatrix} e_1^T \\ e_2^T \\ e_4^T \\ e_8^T \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Le vecteur non redondant $U_3(n)$ est alors tel que :

$$U_3(n) = (u^3(n) \quad u^2(n)u(n-1) \quad u(n)u^2(n-1) \quad u^3(n-1))^T$$

1.2.2 Ecriture matricielle d'un modèle homogène

Considérons le modèle homogène d'ordre p de sortie y_p suivant :

$$y_p(n) = \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \cdots \sum_{n_p=n_{p-1}}^{M-1} h_p(n_1, n_2, \dots, n_p) \prod_{j=1}^p u(n - n_j) \quad (1.12)$$

Il est toujours possible d'écrire cette relation sous la forme matricielle suivante [104, 105] :

$$y_p(n) = U_l^T(n) \mathbf{H}_{l,p-l} U_{p-l}(n), \quad 1 \leq l \leq p-1 \quad (1.13)$$

où $U_p(n)$ est un vecteur de dimension $\frac{(M+p-1)!}{p!(M-1)!}$ qui contient tous les produits d'ordre p du signal u , de la forme suivante :

$$u(n - n_1)u(n - n_2) \cdots u(n - n_p), \quad 0 \leq n_1 \leq n_2 \cdots \leq n_p \leq M-1$$

$\mathbf{H}_{l,p-l}$ est une matrice de dimension $\frac{(M+l-1)!}{l!(M-1)!} \times \frac{(M+p-l-1)!}{(p-l)!(M-1)!}$ telle que son (i, j) -ième élément est le coefficient du noyau qui pondère le produit de la i -ième composante de $U_l(n)$ et de la j -ième composante de $U_{p-l}(n)$ dans (1.12). Le modèle initial (1.12) étant non redondant certaines composantes de $\mathbf{H}_{l,p-l}$ sont nulles.

Dans le cas d'un noyau quadratique nous obtenons la formulation suivante :

$$\mathbf{H}_{1,1} = \begin{pmatrix} h_2(0,0) & h_2(0,1) & \cdots & h_2(0,M-1) \\ 0 & h_2(1,1) & \cdots & h_2(1,M-1) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_2(M-1,M-1) \end{pmatrix}$$

Par suite, la sortie du modèle homogène quadratique s'écrit :

$$\bar{h}_2[u(n)] = U_1^T(n) \mathbf{H}_{1,1} U_1(n)$$

Il est à noter que dans le cas des noyaux d'ordre supérieur à deux, cette formulation matricielle n'est pas unique. Pour des noyaux pairs, nous pouvons toujours choisir une représentation du type $\mathbf{H}_{p/2,p/2}$, i.e. une matrice carrée.

Exemple 1.3 Prenons le cas d'un noyau cubique de mémoire $M = 3$. Nous avons les deux représentations matricielles suivantes :

$$y_3(n) = U_2^T(n) \mathbf{H}_{2,1} U_1(n) \quad \text{et} \quad y_3(n) = U_1^T(n) \mathbf{H}_{1,2} U_2(n) = U_2^T(n) \mathbf{H}_{1,2}^T U_1(n)$$

On a donc $\mathbf{H}_{1,2} = \mathbf{H}_{2,1}^T$, avec :

$$\begin{aligned} U_1(n) &= \begin{pmatrix} u(n) & u(n-1) & u(n-2) \end{pmatrix}^T \\ U_2(n) &= \begin{pmatrix} u^2(n) & u(n)u(n-1) & u(n)u(n-2) & u^2(n-1) \\ & u(n-1)u(n-2) & u^2(n-2) \end{pmatrix}^T \\ \mathbf{H}_{2,1} &= \begin{pmatrix} h_3(0,0,0) & h_3(0,0,1) & h_3(0,0,2) \\ 0 & h_3(0,1,1) & h_3(0,1,2) \\ 0 & 0 & h_3(0,2,2) \\ 0 & h_3(1,1,1) & h_3(1,1,2) \\ 0 & 0 & h_3(1,2,2) \\ 0 & 0 & h_3(2,2,2) \end{pmatrix} \end{aligned}$$

D'une manière générale, la sortie d'un modèle de Volterra d'ordre P peut donc s'écrire comme suit :

$$y(n) = U_1^T(n) \mathbf{H}_1 + \sum_{p=2}^P U_{l_p}^T(n) \mathbf{H}_{l_p, p-l_p} U_{p-l_p}(n) \quad (1.14)$$

1.2.3 La décomposition en coordonnées diagonales

Rappelons l'expression du modèle homogène d'ordre p suivant :

$$y_p(n) = \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \cdots \sum_{n_p=n_{p-1}}^{M-1} h_p(n_1, n_2, \dots, n_p) \prod_{j=1}^p u(n - n_j)$$

Via quelques manipulations algébriques, il est possible de voir ce filtre de Volterra comme une mise en parallèle de plusieurs filtres linéaires de type *RIF*. Cette transformation, introduite par [112], est aussi appelée implémentation multicanal du filtre de Volterra. Elle a été mise à profit par [124] dans une application en annulation de bruit et par [45, 113] pour l'égalisation aveugle d'un canal non-linéaire modélisé à l'aide des filtres de Volterra.

La décomposition en coordonnées diagonales est obtenue en faisant le changement de variables suivant dans l'équation du modèle homogène rappelé ci-dessus :

$$n_j = m + i_{j-1}, \quad j = 1, \dots, p \quad \text{avec} \quad i_0 = 0$$

D'où :

$$y_p(n) = \sum_{m=0}^{M-1} \sum_{i_1=0}^{M-1-m} \sum_{i_2=i_1}^{M-1-m} \cdots \sum_{i_{p-1}=i_{p-2}}^{M-1-m} h_p(m, m + i_1, \dots, m + i_{p-1}) \\ \times u(n-m) \prod_{j=1}^{p-1} u(n-m-i_j)$$

En intervertissant l'ordre des sommations, on obtient :

$$y_p(n) = \sum_{i_1=0}^{M-1} \sum_{i_2=i_1}^{M-1} \cdots \sum_{i_{p-1}=i_{p-2}}^{M-1} \left\{ \sum_{m=0}^{M-1-i_{p-1}} h_p(m, m + i_1, \dots, m + i_{p-1}) u(n-m) \prod_{j=1}^{p-1} u(n-m-i_j) \right\} \\ = \sum_{i_1=0}^{M-1} \sum_{i_2=i_1}^{M-1} \cdots \sum_{i_{p-1}=i_{p-2}}^{M-1} h_p(n, n + i_1, \dots, n + i_{p-1}) * u_{i_1, i_2, \dots, i_{p-1}}(n) \quad (1.15)$$

où le symbole $*$ représente la convolution linéaire et $u_{i_1, i_2, \dots, i_{p-1}}(n)$ le signal donné par :

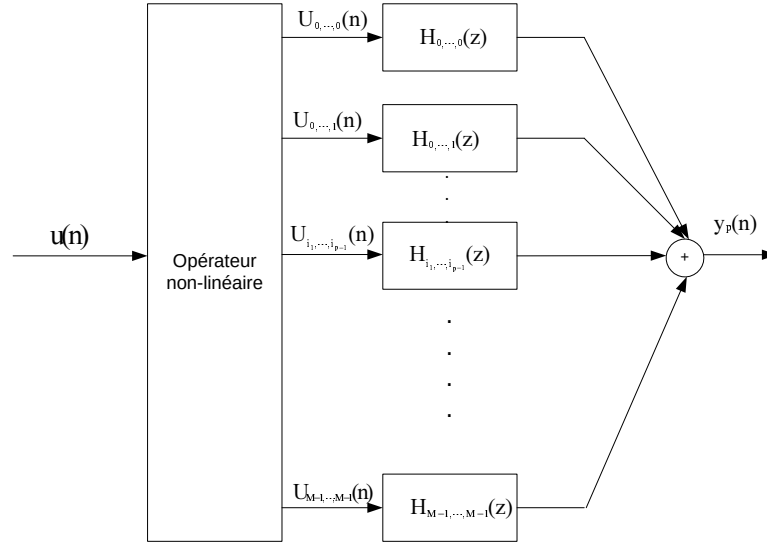
$$u_{i_1, i_2, \dots, i_{p-1}}(n) = u(n) \prod_{j=1}^{p-1} u(n - i_j)$$

En appliquant la transformée en z sur l'équation (1.15), on a :

$$Y_p(z) = \sum_{i_1=0}^{M-1} \sum_{i_2=i_1}^{M-1} \cdots \sum_{i_{p-1}=i_{p-2}}^{M-1} H_{i_1, \dots, i_{p-1}}(z) U_{i_1, \dots, i_{p-1}}(z) \quad (1.16)$$

où $U_{i_1, \dots, i_{p-1}}(z)$ et $Y_p(z)$ représentent les transformées en z respectives de l'entrée $u_{i_1, \dots, i_{p-1}}(n)$ et de la sortie $y_p(n)$ et $H_{i_1, \dots, i_{p-1}}(z)$ la fonction de transfert du filtre RIF de mémoire $M - i_{p-1}$, telle que :

$$H_{i_1, \dots, i_{p-1}}(z) = \sum_{m=0}^{M-1-i_{p-1}} h_p(m, m + i_1, \dots, m + i_{p-1}) z^{-m}$$


 FIG. 1.1 – Implémentation multicanal d'un modèle de Volterra homogène d'ordre p

Le nombre total de filtres linéaires $H_{i_1,\dots,i_{p-1}}$, $i_j = i_{j-1}, \dots, M-1$, $j = 1, \dots, p-1$, $i_0 = 0$, nécessaires pour une implémentation multicanal est de $\frac{(p+M-2)!}{(p-1)!(M-1)!}$ [112].

On peut remarquer que, dans le cas quadratique, la *décomposition en coordonnées diagonales* consiste à former chaque filtre RIF à partir des diagonales de la matrice $\mathbf{H}_{1,1}$.

1.3 Modèles de Volterra simplifiés

Dans cette section nous allons présenter différentes structures qui sont des cas particuliers du modèle de Volterra. Certaines sont obtenues en faisant l'hypothèse de séparabilité du noyau, i.e.

$$h_p(n_1, \dots, n_p) = \prod_{j=1}^p h_1(n_j)$$

1.3.1 Modèles de Wiener et de Hammerstein

Ces modèles sont obtenus en mettant en cascade un modèle dynamique linéaire et un bloc non-linéaire statique. Lorsque la non-linéarité statique suit le bloc dynamique linéaire, le modèle est dit modèle de Wiener; dans le cas contraire, on parle de modèle de Hammerstein. Lorsque le bloc non-linéaire statique est mis en sandwich entre deux blocs dynamiques linéaires, on parle alors de modèle de Wiener-Hammerstein.

Ce type de modèle a été utilisé dans des domaines très variés allant de la biologie à la chimie ou à la psychologie ou la sociologie. [50] cite différentes références ayant trait

à ce type d'application. Dans le domaine des communications, le modèle de Wiener est utilisé pour la modélisation du canal de transmission radio via la fibre optique [39] et le modèle de Hammerstein pour modéliser l'égaliseur dans ce type de communications [40]. On modélise souvent la chaîne de transmission satellitaire par un modèle de Wiener-Hammerstein. On trouvera aussi des applications en filtrage adaptatif dans [10].

1.3.1.1 Modèle de Wiener

Pour une non-linéarité statique polynômiale d'ordre P , la relation d'entrée-sortie de ce modèle est obtenue en considérant :

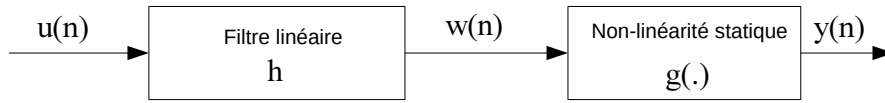


FIG. 1.2 – Modèle de Wiener

$$w(n) = \sum_{n_1=0}^{M-1} h(n_1)u(n - n_1)$$

$$\begin{aligned}
 y(n) &= g[w(n)] = \sum_{p=0}^P c_p w^p(n) \\
 &= c_0 + \sum_{p=1}^P c_p \left(\sum_{n_1=0}^{M-1} h(n_1)u(n - n_1) \right)^p \\
 &= c_0 + \sum_{p=1}^P \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=0}^{M-1} c_p \prod_{j=1}^p h(n_j)u(n - n_j) \\
 &= c_0 + \sum_{p=1}^P \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=0}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j)
 \end{aligned}$$

avec

$$h_p(n_1, \dots, n_p) = c_p \prod_{j=1}^p h(n_j)$$

Le modèle de Wiener est donc un modèle de Volterra à noyaux séparables.

1.3.1.2 Modèle de Hammerstein

Le modèle de Hammerstein peut être vu comme étant le dual du modèle de Wiener composé des mêmes blocs mais placés dans un ordre inversé. On a :

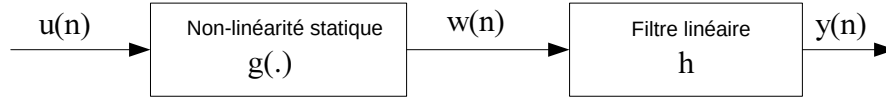


FIG. 1.3 – Modèle de Hammerstein

$$w(n) = g[u(n)] = \sum_{p=0}^P c_p u^p(n)$$

$$\begin{aligned}
 y(n) &= \sum_{n_1=0}^{M-1} h(n_1) w(n-n_1) \\
 &= c_0 \sum_{n_1=0}^{M-1} h(n_1) + \sum_{p=1}^P c_p \sum_{n_1=0}^{M-1} h(n_1) u^p(n-n_1) \\
 &= h_0 + \sum_{p=1}^P \bar{h}_p[u(n)]
 \end{aligned}$$

avec

$$\begin{aligned}
 h_0 &= c_0 \sum_{n_1=0}^{M-1} h(n_1) \\
 h_p(n_1, \dots, n_p) &= \begin{cases} c_p h(m), & n_1 = \dots = n_p = m \\ 0 & \text{sinon} \end{cases}, \quad 1 \leq p \leq P
 \end{aligned}$$

Le modèle de Hammerstein correspond à un modèle de Volterra diagonal, ce qui en fait un cas très particulier.

1.3.2 Modèle de Volterra quadratique à q diagonales

Dans la section précédente nous avons montré que le modèle de Hammerstein correspond à un modèle de Volterra diagonal. Dans le cas quadratique, il est aussi possible de tronquer le modèle général aux q premières diagonales proches de la diagonale principale. Ce modèle a été suggéré par [37] dans le contexte de l'annulation d'écho. Le modèle classique d'un chemin d'écho est un bloc non-linéaire suivi d'un modèle linéaire. Sur la base de mesures expérimentales, il a été montré que la non-linéarité ne doit pas être statique. En représentant ce modèle par un modèle de Volterra quadratique, on s'aperçoit que les coefficients du noyau les plus significatifs sont ceux qui se trouvent sur la diagonale principale et sur les diagonales qui lui sont voisines lorsqu'on considère la représentation matricielle du noyau quadratique. Par conséquent, on a :

$$y(n) = \sum_{i=1}^q \sum_{j=0}^{M-1-i} h(j, j+i-1) u(n-j) u(n-j-i+1)$$

où $q < M$ est le nombre de diagonales utilisées. Ce modèle s'obtient en fait en utilisant la décomposition en coordonnées diagonales décrites dans le paragraphe 1.2.3.

1.3.3 Modèle parallèle-cascade

Palm [102] a montré qu'un modèle de Volterra tronqué, de mémoire finie peut être exactement représenté par une mise en parallèle d'un nombre fini de modèles de Wiener-Hammerstein. Par la suite, Korenberg [71] a montré que cela était aussi vrai pour une mise en parallèle de modèles de Wiener. L'apport de Korenberg est considérable car l'identification d'un modèle de Wiener est beaucoup plus simple que celle d'un modèle de Wiener-Hammerstein.

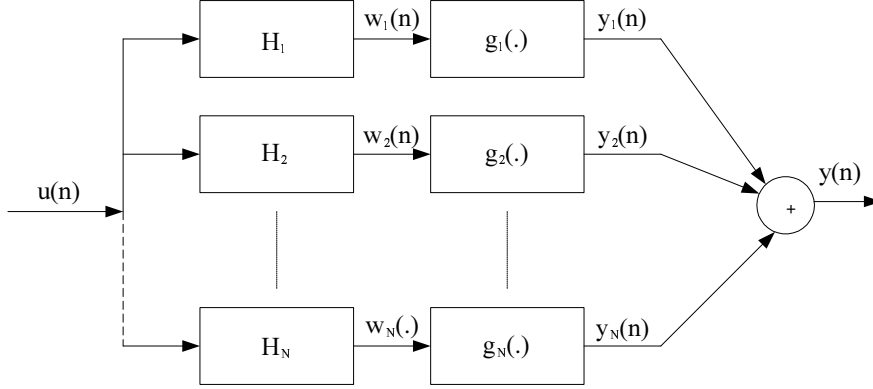


FIG. 1.4 – Modèle parallèle-cascade

1.4 Réduction de complexité par décomposition matricielle ou tensorielle

1.4.1 Décomposition matricielle

Considérons le modèle homogène (1.12) et sa représentation matricielle (1.13). Si r est le rang de $\mathbf{H}_{l,p-l}$, en utilisant sa décomposition en valeurs singulières, on peut donc l'exprimer sous la forme [48] :

$$\mathbf{H}_{l,p-l} = \sum_{i=1}^r \sigma_i \mathbf{W}_i \mathbf{V}_i^T \quad (1.17)$$

où les σ_i sont les valeurs singulières de $\mathbf{H}_{l,p-l}$ et les $\mathbf{W}_i$ et $\mathbf{V}_i$ les vecteurs singuliers à gauche et à droite respectivement. Le rang r est tel que :

$$1 \leq r \leq \min \left(\frac{(M+l-1)!}{l!(M-1)!}, \frac{(M+p-l-1)!}{(p-l)!(M-1)!} \right)$$

La relation (1.13) devient :

$$\begin{aligned} y_p(n) &= \sum_{i=1}^r \sigma_i [U_l^T(n) W_i] [V_i^T U_{p-l}(n)] \\ &= \sum_{i=1}^r \sigma_i y_{l,i}(n) y_{p-l,i}(n) \end{aligned} \quad (1.18)$$

où :

$$y_{l,i}(n) = U_l^T(n) W_i, \quad y_{p-l,i}(n) = V_i^T U_{p-l}(n)$$

Dans le cas d'un noyau d'ordre pair, la matrice $\mathbf{H}_{p/2,p/2}$ est carrée. Nous pouvons donc lui appliquer une décomposition LDL^T telle que :

$$\mathbf{H}_{p/2,p/2} = \sum_{i=1}^r \sigma_i L_i L_i^T \quad (1.19)$$

d'où :

$$y_p(n) = \sum_{i=1}^r \sigma_i (y_{p/2,i}(n))^2, \quad y_{p/2,i} = U_{p/2}^T(n) L_i \quad (1.20)$$

De ce type de décomposition découle une représentation de type parallèle-cascade [25, 104]. Une version adaptative a été proposée par [105].

Chaque branche i est formée à partir des vecteurs W_i , à $\frac{(M+l-1)!}{l!(M-1)!}$ composantes, et V_i , à $\frac{(M+p-l-1)!}{(p-l)!(M-1)!}$ composantes, et de la valeur singulière σ_i . Ce qui fait, pour chaque branche, un nombre de paramètres égal à :

$$\frac{(M+l-1)!(p-l)! + (M+p-l-1)!!}{l!(p-l)!(M-1)!} + 1$$

En conséquence la complexité globale est de :

$$r \left(\frac{(M+l-1)!(p-l)! + (M+p-l-1)!!}{l!(p-l)!(M-1)!} + 1 \right) \quad (1.21)$$

Cette complexité peut être réduite en écartant les valeurs singulières les moins significatives. Si l'on ne considère que les $k < r$ valeurs singulières les plus significatives, la complexité équivalente est :

$$C^{VSD}(k) = k \left(\frac{(M+l-1)!(p-l)! + (M+p-l-1)!!}{l!(p-l)!(M-1)!} + 1 \right) \quad (1.22)$$

Notons par ailleurs que pour un modèle non homogène, il suffit d'utiliser les décompositions de chaque terme homogène. Dans ce cas, la relation (1.14) devient :

$$y(n) = U_1^T(n) H_1 + \sum_{p=2}^P \sum_{i_p=1}^{k_p} \sigma_{i_p} y_{l_p,i_p}(n) y_{p-l_p,i_p}(n) \quad (1.23)$$

où :

$$y_{l_p,i_p}(n) = U_{l_p}^T(n) W_{i_p}, \quad y_{p-l_p,i_p}(n) = V_{i_p}^T U_{p-l_p}(n)$$

1.4.2 Décomposition tensorielle

Les coefficients d'un noyau de Volterra d'ordre $p > 2$ peuvent être regroupés dans un tenseur du même ordre. Il est donc possible d'appliquer sur ce noyau les techniques de décomposition tensorielle en vue de réduire le nombre de paramètres qui le caractérise. Une des décompositions possibles est le modèle PARAFAC (voir l'annexe A.2) introduit par Harshman [56] et Carroll et Chang [23] dans les années soixante-dix. Dans le cas d'un tenseur d'ordre trois, elle consiste à représenter ses éléments par une somme finie de produits d'éléments de trois matrices. Khouaja *et al.* ont appliqué cette décomposition à un noyau de Volterra cubique [64]. Cette décomposition permet d'écrire le noyau cubique tel que :

$$h_3(n_1, n_2, n_3) = \sum_{i=1}^R a(n_1, i) b(n_2, i) c(n_3, i), \quad n_j = 0, \dots, M-1, \quad j = 1, 2, 3 \quad (1.24)$$

où R est le nombre de facteurs de la décomposition PARAFAC et les $a(n_1, i)$, $b(n_2, i)$, $c(n_3, i)$ des éléments des matrices de cette décomposition. Par suite le terme cubique du modèle de Volterra-PARAFAC s'écrit :

$$\begin{aligned} y_3(n) &= \sum_{n_1=0}^{M-1} \sum_{n_2=0}^{M-1} \sum_{n_3=0}^{M-1} \left(\sum_{i=1}^R a(n_1, i) b(n_2, i) c(n_3, i) \right) \prod_{j=1}^3 u(n - n_j) \\ &= \sum_{i=1}^R \left(\sum_{n_1=0}^{M-1} a(n_1, i) u(n - n_1) \right) \left(\sum_{n_2=0}^{M-1} b(n_2, i) u(n - n_2) \right) \left(\sum_{n_3=0}^{M-1} c(n_3, i) u(n - n_3) \right) \end{aligned} \quad (1.25)$$

Ce modèle admet aussi une représentation de type parallèle-cascade. Sa complexité paramétrique est [64] :

$$C^{VP} = 3MR \quad (1.26)$$

Dans le cas particulier d'un noyau séparable, $R = 1$ suffit pour représenter le noyau avec une très bonne précision.

Remarque 1.1 Les modèles résultant d'une décomposition matricielle ou tensorielle conduisent à des implémentations de type parallèle-cascade. Ces structures étant modulaires, elles sont bien indiquées pour l'implémentation VLSI des filtres de Volterra [105, 62]. Cependant, le gros inconvénient est la perte de la linéarité par rapport aux coefficients du modèle. En conséquence, les algorithmes d'estimation sont généralement sous-optimaux. Ce problème est aussi rencontré dans les approximations de modèle de Volterra proposées par [43, 62].

1.5 Estimation au sens de l'erreur quadratique moyenne minimale d'un modèle de Volterra

Considérons le problème de l'estimation des paramètres d'un modèle de Volterra représentant un système ayant $d(n)$ comme signal désiré et $u(n)$ comme signal d'entrée.

$$y(n) = \sum_{p=1}^P \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=n_{p-1}}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j)$$

Les noyaux étant considérés dans leur forme triangulaire, nous avons la représentation vectorielle (1.11) :

$$y(n) = \Theta^T U(n)$$

où le vecteur de régression $U(n)$ et le vecteur Θ sont définis comme :

$$U(n) = (U_1^T(n) \cdots U_P^T(n))^T, \quad \Theta = (\Theta_1^T \cdots \Theta_P^T)^T$$

$U_p(k)$ et Θ_p étant les vecteurs de régression et de paramètres relatifs au terme homogène d'ordre p .

Les signaux d'entrée-sortie $u(n)$ et $d(n)$ sont supposés être des signaux aléatoires à valeurs réelles, conjointement stationnaires au moins jusqu'à l'ordre $2P$. La méthode d'estimation envisagée consiste à minimiser l'erreur quadratique moyenne (EQM) d'estimation par rapport au vecteur de paramètres Θ :

$$\min_{\Theta} J_{EQM}(\Theta) = \mathbb{E} \left[(d(n) - \Theta^T U(n))^2 \right] \quad (1.27)$$

Cette fonction objectif étant quadratique en Θ , elle admet un extremum global unique qui est un minimum donné par :

$$\Gamma_U \Theta_{EQM} = \mathfrak{C}_{dU} \quad (1.28)$$

où $\Gamma_U = \mathbb{E}[U(n)U^T(n)]$ est la matrice d'autocorrélation du vecteur de régression $U(n)$, appelée aussi matrice des moments de l'entrée, $\mathfrak{C}_{dU}$ est le vecteur d'intercorrélacion entre le processus $d(n)$ et le vecteur de régression $U(n)$. Ce système d'équations algébriques linéaires, appelées équations de Wiener-Hopf ou équations normales étendues, constitue la *solution de Wiener*. Lorsque la matrice d'autocorrélation Γ_U est inversible, la séquence d'entrée $\{u(n)\}$ est dite *excitante de façon persistante*, de degré M et d'ordre P , et le vecteur de paramètres estimés peut alors être calculé à l'aide de la formule suivante :

$$\Theta_{EQM} = \Gamma_U^{-1} \mathfrak{C}_{dU} \quad (1.29)$$

On obtient ainsi une formule d'estimation identique à celle du cas linéaire ; seule la définition du vecteur de régression $U(n)$ est différente. Pour l'estimation paramétrique d'un modèle linéaire *RIF*, les quantités Γ_U et $\mathfrak{C}_{dU}$ font intervenir des statistiques du second ordre des signaux d'entrée-sortie, tandis que dans le cas du modèle de Volterra ces quantités renferment des statistiques d'ordres 2 à $2P$ de ces mêmes signaux d'entrée-sortie.

1.5.1 Condition d'inversibilité de la matrice d'autocorrélation du vecteur de régression

La séquence d'entrée $\{u(n)\}$ étant supposée distribuée de manière indépendante et identique (i.i.d., *independent identically distributed*), on a la propriété suivante [93] :

Théorème 1.1 *La matrice d'autocorrélation $\mathbf{\Gamma}_U$ du vecteur de régression $U(n)$ à composantes polynômiales de degré au plus égal à P , en l'entrée $u(n)$ supposée i.i.d., est inversible si et seulement si l'entrée prend au moins $(P + 1)$ valeurs distinctes.*

De ce théorème on peut déduire les résultats suivants [34] :

- Pour un canal de communication numérique non linéaire modélisé à l'aide d'un filtre de Volterra d'ordre P , une séquence d'entrée aléatoire i.i.d. du type $PAM - M$, pouvant prendre M (M étant pair) valeurs distinctes, satisfait la condition de persistance d'excitation (PE) si et seulement si $M \geq P + 1$. En particulier, pour un canal linéaire ($P = 1$) on doit avoir $M \geq 2$, et pour un canal quadratique ($P = 2$) ou cubique ($P = 3$) il faut $M \geq 4$.
- Une séquence d'entrée i.i.d. gaussienne satisfait la condition PE pour tout canal non linéaire d'ordre P fini.
- Toute séquence d'entrée i.i.d., de moyenne nulle et de variance σ^2 , satisfait la condition PE pour tout modèle linéaire RIF, indépendamment de sa distribution. En effet, dans le cas d'un modèle RIF de mémoire M , pour une séquence d'entrée i.i.d. stationnaire et centrée on a $\mathbf{\Gamma}_U = \sigma^2 \mathbf{I}_M$ où $\mathbf{I}_M$ est la matrice identité d'ordre M .

1.5.2 Cas d'un modèle de Volterra du second ordre

Dans ce cas, les équations de Wiener-Hopf (1.28) s'écrivent sous la forme partitionnée suivante :

$$\begin{pmatrix} \mathbf{\Gamma}_{U_1} & \mathbb{E}[U_1 U_2^T] \\ \mathbb{E}[U_2 U_1^T] & \mathbf{\Gamma}_{U_2} \end{pmatrix} \begin{pmatrix} \Theta_{1,EQM} \\ \Theta_{2,EQM} \end{pmatrix} = \begin{pmatrix} \mathfrak{C}_{dU_1} \\ \mathfrak{C}_{dU_2} \end{pmatrix} \quad (1.30)$$

Notons que les blocs diagonaux $\mathbf{\Gamma}_{U_1}$ et $\mathbf{\Gamma}_{U_2}$ contiennent respectivement des moments d'ordres deux et quatre de l'entrée, tandis que les blocs hors diagonaux sont formés de moments d'ordre trois de l'entrée. Supposons que :

- i) La séquence u est centrée ; i.e. $\mathbb{E}\{u\} = 0$.
- ii) Elle suit une distribution de probabilité symétrique. Ses moments d'ordre trois sont donc nuls :

$$\mathbb{E}\{u(n - \tau)u(n - \tau_1)u(n - \tau_2)\} = 0, \forall \tau, \tau_1, \tau_2 = 0, 1, \dots, M - 1$$

La relation (1.30) devient alors :

$$\begin{pmatrix} \mathbf{\Gamma}_{U_1} & 0 \\ 0 & \mathbf{\Gamma}_{U_2} \end{pmatrix} \begin{pmatrix} \Theta_{1,EQM} \\ \Theta_{2,EQM} \end{pmatrix} = \begin{pmatrix} \mathfrak{C}_{dU_1} \\ \mathfrak{C}_{dU_2} \end{pmatrix}$$

La solution optimale eut alors être décomposée comme suit :

$$\mathbf{\Gamma}_{U_1} \Theta_{1,EQM} = \mathfrak{C}_{dU_1} \quad (1.31)$$

$$\mathbf{\Gamma}_{U_2} \Theta_{2,EQM} = \mathfrak{C}_{dU_2} \quad (1.32)$$

A partir de l'équation (1.31) nous pouvons conclure que, le vecteur de paramètres estimés relatif au terme homogène d'ordre un est identique à la solution optimale du filtre linéaire de Wiener. Si en plus le signal est supposé *i.i.d.* alors $\mathbf{\Gamma}_{U_1} = \sigma^2 \mathbf{I}_M$. Il s'en suit que :

$$\Theta_{1,EQM} = \frac{1}{\sigma^2} \mathfrak{C}_{dU_1}$$

1.5.3 Cas d'un modèle de Volterra cubique

Nous venons de voir que, dans le cas d'un modèle de Volterra linéaire-quadratique, l'hypothèse d'une distribution symétrique de l'entrée permet de découpler l'estimation des noyaux linéaire et quadratique. Toutefois, cette hypothèse n'implique pas le découplage de l'estimation des noyaux d'un modèle de Volterra d'ordre supérieur à deux. Ainsi, pour un modèle cubique l'estimation des parties linéaire et cubique est couplée. Cependant, il a été montré [131] que pour une séquence *i.i.d.* il est possible d'estimer les coefficients des noyaux de Volterra sans aucun calcul matriciel, réduisant ainsi la complexité de calcul de la procédure d'estimation. Il a aussi été montré que pour des signaux d'entrées complexes pouvant s'écrire sous la forme $u = u_r + ju_i$, où u_r et u_i sont des variables aléatoires réelles, indépendantes et symétriquement distribuées, et $j^2 = -1$, le découplage est aussi réalisé [108]. Les signaux de type QAM appartiennent à cette famille. Le découplage est aussi assuré lorsque les signaux sont de type PSK [128].

1.6 Estimation adaptative

Dans le cas où le système nécessite un traitement de données en temps réel ou s'il comporte d'éventuelles non stationnarités, l'implémentation récursive des algorithmes d'estimation s'impose. Généralement, en raison de la propriété de linéarité du modèle de Volterra par rapport à ses paramètres, les algorithmes adaptatifs utilisés dans le cas des filtres linéaires peuvent être étendus aux filtres de Volterra.

1.6.1 Algorithmes de type LMS

Il est mis en œuvre en minimisant le critère de l'erreur quadratique moyenne. A chaque instant n , le vecteur des paramètres est adapté suivant la relation suivante :

$$\Theta_p(n+1) = \Theta_p(n) + \mu_p \varepsilon(n) U_p(n) \quad (1.33)$$

où μ_p est une petite constante positive qui règle la vitesse de convergence de l'algorithme et $\varepsilon(n) = d(n) - \Theta^T U(n)$. Sous certaines hypothèses simplificatrices, on peut montrer que la convergence de l'algorithme LMS est garantie lorsque [86]

$$0 < \mu_p < \frac{2}{\lambda_{\max}}, \quad \forall p$$

où $\lambda_{\max}$ est la valeur propre maximale de la matrice des moments de l'entrée. Comme dans le cas linéaire, la vitesse de convergence de l'algorithme dépend du ratio entre la valeur propre maximale et la valeur propre minimale de la matrice des moments de l'entrée. La convergence est d'autant plus rapide que ce ratio, qui est le nombre condition de la matrice des moments, est faible [86, 57]. La complexité de calcul de cet algorithme est de $O(M^P)$.

En vue d'améliorer la vitesse de convergence de l'algorithme LMS et d'être moins dépendant vis à vis des statistiques du signal d'entrée, on substitue à la structure transversale du filtre une structure en treillis. L'objectif est d'orthogonaliser les signaux d'entrée et donc d'estimer la réponse désirée comme étant une combinaison linéaire de signaux orthogonaux. Plusieurs études ont été menées dans ce sens. Entre autres, nous pouvons citer [70, 86]. D'autres algorithmes de type LMS ont été proposés par [52, 74].

1.6.2 Algorithmes de type RLS

L'algorithme RLS trouve la solution analytique d'un problème d'optimisation formulé dans un cadre déterministe [86]. Pour ce faire, on minimise la fonction de coût

$$J(n) = \sum_{m=0}^n \lambda^{n-m} (d(n) - \Theta^T U(n))^2$$

où λ est le facteur d'oubli pris entre 0 et 1. La solution optimale à la minimisation de ce critère est donné par :

$$\hat{\Phi}_U(n) \Theta(n) = \hat{\mathcal{C}}_{dU}(n)$$

où :

$$\begin{aligned} \hat{\Phi}_U(n) &= \sum_{m=0}^n \lambda^{n-m} U(m) U^T(m) = \lambda \hat{\Phi}_U(n-1) + U(n) U^T(n) \\ \hat{\mathcal{C}}_{dU}(n) &= \sum_{m=0}^n \lambda^{n-m} d(m) U(m) = \lambda \hat{\mathcal{C}}_{dU}(n-1) + d(n) U(n) \end{aligned}$$

Pour estimer $\Theta(n)$, il est nécessaire de calculer l'inverse de la matrice $\hat{\Phi}_U(n)$. Grâce aux formulations récursives ci-dessus et au lemme de l'inversion matricielle, nous avons :

$$\hat{\Phi}_U^{-1}(n) = \lambda^{-1} \hat{\Phi}_U^{-1}(n-1) - \lambda^{-1} \kappa(n) U^T(n) \hat{\Phi}_U^{-1}(n-1) \quad (1.34)$$

avec

$$\kappa(n) = \frac{\lambda^{-1} \hat{\Phi}_U^{-1}(n-1) U(n)}{1 + \lambda^{-1} U^T(n) \hat{\Phi}_U^{-1}(n-1) U(n)} \quad (1.35)$$

Par suite, le vecteur des paramètres optimaux est récursivement obtenu comme suit :

$$\Theta(n) = \Theta(n-1) + \kappa(n)\varepsilon(n) \quad (1.36)$$

où $\varepsilon(n) = d(n) - \Theta^T(n-1)U(n)$.

Les conditions initiales de cet algorithmes sont telles que : $\Theta(0) = 0$ et $\hat{\Phi}_U^{-1}(0) = \gamma^{-1}\mathbf{I}$, où γ est une petite constante positive.

L'algorithme RLS converge généralement plus vite que le LMS et ce de manière indépendante des statistiques du signal d'entrée [86]. Cependant, sa complexité de calcul est bien plus élevée : elle est de $O(M^{2P})$. Des approches de développement d'algorithmes rapides sont présentes dans la littérature mais en général leur coût de calcul reste plus élevé que celui du LMS [29, 77, 123, 101].

1.7 Exemples de canaux de communication non-linéaires

Nous présentons dans ce paragraphe des caractéristiques de quelques canaux de communications non-linéaires rencontrés en pratique. Dans le chapitre 5, nous détaillerons un autre système de communication non-linéaire, celui de l'accès au réseau sans fil via une liaison par fibre optique.

1.7.1 Communication par satellite

Afin d'obtenir une bonne puissance de transmission, les amplificateurs (généralement des TWTA pour *Travelling Wave Tube Amplifier* en anglais) du transpondeur localisé dans le satellite et de la station terrienne sont utilisés près de leur niveau de saturation. Il en résulte une distorsion non-linéaire en sortie de l'amplificateur. Un amplificateur radio-fréquence (RF) opérant dans sa zone de fonctionnement non-linéaire présente deux types de non-linéarité. La première lie la puissance du signal d'entrée à la sortie de l'amplificateur. Ce type de distorsion est appelé conversion AM/AM. La seconde lie la phase du signal de sortie avec la puissance du signal d'entrée. Elle est appelée conversion AM/PM. Les modèles les plus couramment utilisés pour modéliser ces effets sont dus à Saleh [118].

Dans une liaison satellitaire classique cette non-linéarité est mise en sandwich entre deux systèmes linéaires invariants dans le temps $h_{up}(t)$ et $h_{down}(t)$ qui modélisent les canaux de propagation de la terre vers le satellite et inversement. Il a été montré que la représentation en bande de base de ce système est précisément décrite par le modèle de Volterra ci-dessous [9] :

$$y(n) = \sum_{p=0}^P \sum_{n_1}^M \cdots \sum_{n_{2p+1}}^M h_{2p+1}(n_1, \dots, n_{2p+1}) \prod_{j=1}^p u^*(n-n_j) \prod_{j=p+1}^{2p+1} u(n-n_j) \quad (1.37)$$

où $u(n)$ est la séquence de symboles du signal émis et $h_{2p+1}(n_1, \dots, n_{2p+1})$ sont les noyaux de Volterra en bande de base déterminés à partir des coefficients des canaux linéaires $h_{up}(t)$ et $h_{down}(t)$ et des coefficients du développement de Taylor des non-linéarités de l'amplificateur. Il est à noter que l'absence des noyaux pairs dans (1.37) est due au fait qu'ils génèrent des composantes spectrales qui sont en dehors de la bande passante du canal et qui sont donc rejetées par le filtre passe-bande suivant l'amplificateur.

1.7.2 Systèmes de communication sans fil à large bande

Les systèmes de communication sans fil à large bande requièrent des modulations permettant une bonne exploitation de la bande passante. C'est le cas des modulations QAM [61]. Cependant, le problème avec ces modulations est que leur enveloppe présente de larges fluctuations qui rendent le système sensible aux non-linéarités de l'amplificateur de puissance (HPA, de l'anglais *high power amplifier*). Les HPA utilisés sont de type TWTA ou SSPA (*Solid State Power Amplifier*) pour des signaux RF ayant une fréquence de 60 GHz. Le canal de propagation étant généralement représenté par un modèle linéaire, le modèle global du canal de transmission formé par la mise en cascade du HPA et du canal linéaire peut être représenté par un modèle de Volterra.

Il est à noter que ce type de représentation peut être fait pour tout type de système de communication utilisant un amplificateur de puissance à la limite de saturation [111]. Les systèmes de transmission OFDM (*Orthogonal Frequency Division Multiplexing*) sont notamment soumis aux distorsions non-linéaires introduites par le HPA [114].

1.7.3 Communication dans un environnement fermé

Considérons le cas d'une personne dans un environnement fermé tel qu'une salle ou un véhicule communiquant avec un téléphone portable. Le haut parleur introduit un comportement non-linéaire alors que le milieu physique entre le haut-parleur et le micro du téléphone peut être représenté par un modèle linéaire. Le canal ainsi décrit est donc globalement non-linéaire. Il est à noter que dans le cas d'une salle, la réponse impulsionnelle dépend de la réverbération et des caractéristiques d'absorption de la salle. Une telle réponse impulsionnelle est généralement très longue générant ainsi un modèle de Volterra équivalent ayant un nombre très élevé de paramètres. Le canal ainsi modélisé peut être utilisé pour faire de l'annulation d'échos [89].

1.8 Egalisation des systèmes de Volterra

Dans plusieurs applications dans lesquelles le système est représenté par un modèle de Volterra il est parfois nécessaire de reconstituer le signal d'entrée à partir du signal de

sortie. Une telle opération nécessite un égaliseur. Les principales méthodes d'égalisation de systèmes de Volterra SISO sont : la théorie de l'inverse d'ordre P , la méthode du point fixe, l'égalisation de type Wiener et la méthode des racines de polynômes. Toutes ces méthodes sont supervisées et requièrent en général la connaissance des coefficients du modèle de Volterra.

1.8.1 Inverse d'ordre P

Contrairement au cas linéaire où on peut définir l'inverse d'un système, ce n'est généralement pas le cas pour les systèmes non-linéaires. Tout au plus, on peut définir un inverse local, i.e. restreint à un intervalle d'amplitude du signal d'entrée. La théorie de l'inverse d'ordre P a été introduite par Schetzen en 1976 [120].

Définition 1.3 *L'inverse d'ordre P d'un système non-linéaire H est un système non-linéaire $\bar{H}$ tel que la mise en cascade des deux systèmes soit représentée par un modèle de Volterra Q dont les noyaux d'ordre 2 à P sont nuls :*

$$Q(u(n)) = u(n) + \sum_{p=P+1}^{\infty} \bar{Q}_p(u(n)) \quad (1.38)$$

où $\bar{Q}_p(u(n))$ est le terme homogène d'ordre p .

Plusieurs auteurs se sont penchés sur la mise en place d'algorithmes permettant de générer cet inverse [119, 22]. L'écueil principal de cette méthode est la non maîtrise des termes d'ordre supérieur de la mise en cascade du système et de son inverse, i.e.

$\sum_{p=P+1}^{\infty} \bar{Q}_p(u(n))$. La convergence d'un égaliseur basé sur cette théorie n'est donc assurée que pour un système non-linéaire tel que ces termes d'ordre supérieur soient peu influents.

1.8.2 Méthode du point fixe

Cette méthode a été appliquée par [92] dans un cas général tandis que [129] l'ont appliquée au cas des systèmes de communication par satellite utilisant des signaux PSK. Plus récemment, elle a été appliquée d'une part aux systèmes SIMO non-linéaires et d'autre part aux systèmes non-linéaires utilisant un précodage [116]. La méthode se base sur le théorème du point fixe de Banach. Ce théorème garantit, pour une transformation contractante, l'existence et l'unicité d'un point fixe, prescrit la manière de le trouver et donne une indication sur la rapidité de convergence de la méthode. Il a été montré qu'en connaissant les coefficients des noyaux de Volterra représentant le canal, sous certaines conditions nécessaires et suffisantes [92, 130], il est possible de formuler le problème de l'égalisation des systèmes de Volterra comme un problème de recherche de point fixe. Une des conditions est la nécessité d'avoir un noyau du premier

ordre du modèle de Volterra à phase minimale. D'une manière générale, la propriété de contraction est difficile à garantir ce qui ne permet pas de garantir la convergence de la méthode.

1.8.3 Egalisation de type Wiener

Le principe de cette méthode est de rechercher l'égaliseur $\bar{H}$ comme étant le filtre de Volterra, dont les coefficients sont contenus dans le vecteur $\bar{H}$, tel que :

$$\bar{H}_{opt} = \arg \min_{\bar{H}} \mathbb{E} \left(\left\| \bar{H}^T Y(n) - u(n - \delta_e) \right\|^2 \right) \quad (1.39)$$

où $Y(n)$ est le vecteur d'observation formé par les échantillons du signal reçu et leurs produits croisés jusqu'à l'ordre de non-linéarité de l'égaliseur et δ_e est le retard d'égalisation. Cette méthode a d'abord été utilisée par [8] dans le cas des systèmes de communication par satellite. Il est à noter qu'après avoir fixé l'ordre et la mémoire de l'égaliseur, l'erreur résultant de la minimisation de l'erreur quadratique moyenne en sortie de l'égaliseur tend à croître avec l'ordre de non-linéarité du système [115].

1.8.4 Méthode des racines de polynômes

Le principe de cette méthode, suggérée par [115], est de former un polynôme en $u(n)$ puis de rechercher ses racines. Le symbole le plus proche d'une des racines est choisi comme étant $u(n)$. Pour ce faire, on suppose connus le modèle du canal, i.e. les coefficients des noyaux de Volterra représentant le canal, et les symboles antérieurs à $u(n)$. On peut donc former un polynôme dont les coefficients dépendent des symboles antérieurs et des coefficients du modèle. Sans perte de généralité, prenons le cas d'un canal représenté par un modèle de Volterra d'ordre trois. On a :

$$\begin{aligned} y(n) = & \sum_{n_1=0}^{M-1} h_1(n_1)u(n-n_1) + \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} h_2(n_1, n_2)u(n-n_1)u(n-n_2) \\ & + \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(n_1, n_2, n_3)u(n-n_1)u(n-n_2)u(n-n_3) \end{aligned}$$

$$\begin{aligned}
 = & h_1(0)u(n) + \sum_{n_1=1}^{M-1} h_1(n_1)u(n-n_1) + h_2(0,0)u^2(n) \\
 & + u(n) \sum_{n_2=1}^{M-1} h_2(0,n_2)u(n-n_2) + \sum_{n_1=1}^{M-1} \sum_{n_2=n_1}^{M-1} h_2(n_1,n_2)u(n-n_1)u(n-n_2) \\
 & + h_3(0,0,0)u^3(n) + u^2(n) \sum_{n_3=1}^{M-1} h_3(0,0,n_3)u(n-n_3) \\
 & + u(n) \sum_{n_2=1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(0,n_2,n_3)u(n-n_2)u(n-n_3) \\
 & + \sum_{n_1=1}^{M-1} \sum_{n_2=n_1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(n_1,n_2,n_3)u(n-n_1)u(n-n_2)u(n-n_3)
 \end{aligned}$$

En regroupant les différents termes, nous obtenons l'équation polynômiale suivante :

$$c_0 + c_1 u(n) + c_2 u^2(n) + c_3 u^3(n) = 0 \quad (1.40)$$

avec :

$$\begin{aligned}
 c_0 &= -y(n) + \sum_{n_1=1}^{M-1} h_1(n_1)u(n-n_1) + \sum_{n_1=1}^{M-1} \sum_{n_2=n_1}^{M-1} h_2(n_1,n_2)u(n-n_1)u(n-n_2) \\
 &+ \sum_{n_1=1}^{M-1} \sum_{n_2=n_1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(n_1,n_2,n_3)u(n-n_1)u(n-n_2)u(n-n_3) \\
 c_1 &= h_1(0) + \sum_{n_2=1}^{M-1} h_2(0,n_2)u(n-n_2) + \sum_{n_2=1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(0,n_2,n_3)u(n-n_2)u(n-n_3) \\
 c_2 &= h_2(0,0) + \sum_{n_3=1}^{M-1} h_3(0,0,n_3)u(n-n_3) \\
 c_3 &= h_3(0,0,0)
 \end{aligned}$$

Le symbole émis est donc choisi comme étant une racine de l'équation (1.40). Les performances de la méthode dépendent fortement de la qualité de l'estimation du modèle et des symboles antérieurs. L'avantage de cette méthode est qu'elle est utilisable sans condition particulière sur le noyau d'ordre un, i.e. qu'il soit à phase minimale ou non. Ses performances sont aussi indépendantes de la nature de la non-linéarité du système. Cependant la nature DFE (*decision feedback equalizer*) de la méthode peut générer une propagation d'erreur. Nous reviendrons sur cette méthode dans le chapitre 5.

1.9 Conclusion

Dans ce chapitre, nous avons décrit le modèle de Volterra et énoncé quelques applications dans le domaine des télécommunications. Deux problèmes majeurs ont été mis en

évidence : la complexité paramétrique du modèle et la complexité de l'estimation au sens de l'erreur quadratique moyenne minimale du modèle. Dans les chapitres suivants nous allons développer des solutions à ces problèmes :

- d'une part des techniques de réduction de complexité basées sur le développement des noyaux de Volterra sur des bases de fonctions orthogonales ;
- d'autre part des méthodes orthogonales d'estimation au sens de l'erreur quadratique moyenne minimale.

Chapitre 2

Les Bases de Fonctions Orthogonales Rationnelles

2.1 Introduction

L'utilisation des fonctions orthogonales dans la représentation des systèmes est une approche communément utilisée tant dans la communauté du traitement du signal que de l'automatique, et ce depuis plusieurs décennies. On peut citer, parmi les précurseurs, les travaux de Takenaka et de Malmquist en 1925 portant sur la synthèse de fonctions orthogonales dans le but d'interpoler des fonctions analytiques à l'intérieur du cercle unité [13, 58]. Mais aussi, ceux de Lee et de Wiener qui de leur côté ont fourni un travail conséquent sur les fonctions de Laguerre publié dans les années trente.

La base orthogonale complète la plus connue est certainement celle des fonctions delta de Dirac. Sa transformée en z engendre les filtres à réponse impulsionnelle finie (RIF) auxquels on recourt dans de nombreuses applications du traitement du signal. Ces filtres présentent l'avantage d'une implémentation simple. Une approximation de toute fonction de transfert d'un système stable peut être réalisée à l'aide d'un modèle RIF, modèle pouvant être mis sous forme d'une régression linéaire. Aussi, l'estimation par la méthode des moindres carrés donne des estimées consistantes si le bruit de mesure est centré et non corrélé avec l'entrée du système [53]. Cependant, l'utilisation des filtres RIF pour la représentation des systèmes à longue réponse impulsionnelle requiert un nombre de paramètres nécessairement élevé. L'erreur d'approximation décroît certes lorsque le nombre de paramètres augmente mais cette approximation devient plus sensible au bruit. D'où l'intérêt de trouver des modèles qui génèrent des approximations précises avec peu de paramètres [135].

Pour réduire la longueur des filtres RIF, l'utilisation des filtres à réponse impulsion-

nelle infinie (RII) peut être envisagée. Cependant, la stabilité de leur version adaptative n'est pas toujours garantie. Les filtres orthogonaux à fonction de transfert rationnelle représentent, quant à eux, un compromis entre les deux classes de filtres sus-cités [59, 107, 138]. En effet, ils permettent d'améliorer le conditionnement numérique des algorithmes utilisés et fournissent un ensemble de paramètres suffisamment parcimonieux lorsque leurs pôles sont proches des dynamiques du système à approcher [91]. Les pôles étant fixés, le modèle résultant est linéaire par rapport à ses paramètres, les coefficients de Fourier. La minimisation du critère de l'erreur quadratique moyenne présente un minimum global unique et les estimés sont non biaisés si le bruit de mesure est non corrélé avec le signal d'entrée [15, 132].

Parmi les bases de filtres orthogonaux à fonction de transfert rationnelle, les bases de filtres de Laguerre et de Kautz sont certainement les plus connues. Les filtres de Laguerre sont caractérisés par un pôle unique, stable et réel. Ils sont adaptés à la modélisation des systèmes dont la dynamique dominante est du premier ordre [1, 44, 53, 84, 85, 87, 96, 126, 125]. Les filtres de Kautz, quant à eux, sont caractérisés par une paire de pôles complexes conjugués. De ce fait, ils sont fortement indiqués pour la représentation de systèmes ayant une dynamique dominante du second ordre [94, 98, 133, 135].

Dans le but de modéliser des dynamiques plus complexes, plusieurs généralisations de bases orthogonales à fonction de transfert rationnelle ont été introduites. Certains auteurs les ont appelées Bases Orthogonales Généralisées (BOG) mais nous retiendrons plutôt l'appellation Base de Fonctions Orthogonales Rationnelles (BFOR). Dans ce chapitre, nous présentons les différentes approches de synthèse des BFOR existant dans la littérature. Ces différentes approches peuvent être réunies via une méthode générale de synthèse. Quelle que soit l'approche considérée, une problématique importante dans la synthèse de ces bases est le choix des pôles. En effet, dans le développement d'un système sur une BFOR, le nombre de paramètres utilisés dépend fortement des pôles choisis. Comme indiqué plus haut, plus les pôles choisis sont proches des dynamiques du système dont on veut réaliser une approximation, plus l'ordre de troncature du développement du système sur la base ainsi construite est réduit. Dans ce chapitre, nous ne considérerons que le cas des systèmes linéaires, stables et causaux. Les problèmes d'identification que nous considérerons seront dans le domaine temporel. Pour l'identification des systèmes à l'aide des BFOR dans le domaine fréquentiel, on peut se référer à [4, 3].

2.2 Méthodes de construction des bases de fonctions orthogonales rationnelles

De manière parallèle, deux approches de construction des BFOR ont été proposées : les représentations équilibrées dans l'espace d'état [58, 15, 134, 31] et la procédure d'orthonormalisation de Gram-Schmidt [90, 4].

2.2.1 Préliminaires

Considérons un système mono-entrée mono-sortie d'ordre r , de fonction de transfert scalaire H . Sa représentation dans l'espace d'état $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ est telle que :

$$\begin{aligned} x(n+1) &= \mathcal{A}x(n) + \mathcal{B}u(n) \\ y(n) &= \mathcal{C}x(n) + \mathcal{D}u(n) \end{aligned} \quad (2.1)$$

Les signaux $u(n)$, $y(n)$ et $x(n)$ représentent respectivement l'entrée, la sortie et l'état du système. Les matrices $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ et $\mathcal{D}$ sont respectivement de dimensions $r \times r$, $r \times 1$, $1 \times r$ et 1×1 . La relation entre la fonction de transfert et la représentation d'état est :

$$H(z) = \mathcal{D} + \mathcal{C}(zI - \mathcal{A})^{-1}\mathcal{B} \quad (2.2)$$

La représentation d'état $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ est stable si toutes les valeurs propres de $\mathcal{A}$ sont strictement à l'intérieur du cercle unité.

On définit aussi les matrices $\mathcal{P}$ et $\mathcal{Q}$, appelées respectivement matrice de contrôlabilité et matrice d'observabilité. Si le système est stable, elles sont solutions des équations de Lyapunov suivantes :

$$\begin{aligned} \mathcal{A}\mathcal{P}\mathcal{A}^H + \mathcal{B}\mathcal{B}^H &= \mathcal{P} \\ \mathcal{A}^H\mathcal{Q}\mathcal{A} + \mathcal{C}^H\mathcal{C} &= \mathcal{Q} \end{aligned}$$

En considérant que H est analytique en dehors du cercle unité, son développement en série de Laurent est :

$$H(z) = \sum_{i=0}^{\infty} h_i z^{-i}$$

On définit alors la matrice de Hankel associée à H par :

$$\mathcal{H}_{ij}(H) = h_{i+j-1}$$

Une représentation d'état stable est dite équilibrée si $\mathcal{P} = \mathcal{Q} = \Sigma$, où Σ est la matrice diagonale des valeurs singulières positives de la matrice de Hankel classées en ordre décroissant. On parle de représentation équilibrée à l'entrée si $\mathcal{P} = I$, $\mathcal{Q} = \Sigma^2$ et de représentation équilibrée en sortie si $\mathcal{P} = \Sigma^2$, $\mathcal{Q} = I$ [58].

2.2.1.1 Filtre passe-tout

En rappelant que pour tout polynôme en z , $P(z) = \sum_{i=0}^{n_p} p_i z^i$, son conjugué est un polynôme en z^{-1} noté $P^*(z^{-1}) = \sum_{i=0}^{n_p} p_i^* z^{-i}$, le filtre de fonction de transfert H est dit *passe-tout* si :

$$H(z)H^*(z^{-1}) = 1 \quad (2.3)$$

Il est, en outre, dit orthogonal si sa représentation dans l'espace d'état (2.1) est équilibrée en entrée et en sortie. Cela signifie que les matrices de contrôlabilité et d'observabilité sont égales à la matrice identité. En conséquence, les valeurs singulières de

la matrice de Hankel associée à un filtre passe-tout orthogonal sont égales à un [58]. La condition d'orthogonalité d'un filtre passe-tout se résume alors par les propriétés suivantes :

$$\mathcal{A}\mathcal{A}^H + \mathcal{B}\mathcal{B}^H = I \quad (2.4)$$

$$\mathcal{A}^H\mathcal{A} + \mathcal{C}^H\mathcal{C} = I \quad (2.5)$$

[117] ont, par ailleurs, montré qu'une condition nécessaire et suffisante pour que le filtre H soit un passe-tout orthogonal est que la matrice F , définie par (2.6) soit unitaire.

$$F = \begin{pmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{pmatrix} \quad (2.6)$$

2.2.1.2 Produit scalaire

On définit l'espace de Hilbert $\ell_2(\mathbb{N})$ des fonctions de carré sommable à support dans $\mathbb{N}$, doté du produit scalaire :

$$\langle b_k, b_l \rangle = \sum_{i \in \mathbb{N}} b_k(i) b_l^*(i) \quad (2.7)$$

Cet espace est aussi appelé espace de Lebesgue. Pour la classe de systèmes traités, la réponse impulsionnelle est une fonction de $\ell_2(\mathbb{N})$.

Considérons à présent l'espace de Hilbert associé aux fonctions de transfert de systèmes discrets, stables et causaux. Rappelons, pour ce faire, que la transformée en z d'une séquence $\{h(i)\} \in \ell_2(\mathbb{Z})$ est :

$$H(z) = \sum_{i \in \mathbb{N}} h(i) z^{-i}$$

Elle appartient à l'espace de Hilbert $L_2(\mathbb{T})$ des fonctions de carré intégrable sur le cercle unité $\mathbb{T}$. On associe à cet espace le produit scalaire suivant :

$$\langle B_k, B_l \rangle = \frac{1}{2\pi j} \oint B_k(z) B_l^*(z^{-1}) \frac{dz}{z} \quad (2.8)$$

Un sous-espace de $L_2(\mathbb{T})$ est l'espace des fonctions de carré intégrable sur le cercle unité qui sont analytiques à l'extérieur du cercle unité, appelé espace de Hardy. Il est à noter que les fonctions obtenues après l'application de la transformée en z sur un ensemble de fonctions orthogonales dans l'espace de Lebesgue sont aussi orthogonales dans l'espace de Hardy ; la transformée en z conserve la propriété d'orthogonalité.

Par ailleurs, on définit le produit scalaire de deux variables aléatoires $w(n)$ et $v(n)$ par :

$$\langle w(n), v(n) \rangle = \mathbb{E}[w(n) v^*(n)] \quad (2.9)$$

où $\mathbb{E}$ est l'opérateur espérance mathématique.

2.2.1.3 Orthogonalité de deux vecteurs aléatoires

Considérons deux vecteurs aléatoires X et Y . On définit les matrices d'intercorrélation Γ_{XY} et d'intercovariance $\mathfrak{C}_{XY}$ par :

$$\Gamma_{XY} = \mathbb{E} [XY^H], \quad \mathfrak{C}_{XY} = \mathbb{E} [(X - \mathbb{E}[X])[Y - \mathbb{E}[Y]]^H]$$

Elles vérifient la relation suivante :

$$\mathfrak{C}_{XY} = \Gamma_{XY} - \mathbb{E}[X] \mathbb{E}[Y]^H$$

Les vecteurs aléatoires X et Y sont dits non corrélés si :

$$\mathfrak{C}_{XY} = 0$$

et orthogonaux si :

$$\Gamma_{XY} = 0$$

Dans le cas où au moins l'un des deux vecteurs est centré, i.e. de moyenne nulle, la non corrélation des vecteurs X et Y est équivalente à leur orthogonalité.

Considérons $R(z)$ et $T(z)$ des fonctions de transfert stables et causales de deux systèmes mono-entrée multi-sorties (SIMO, de l'anglais Single Input Multiple Outputs) tels que $W(n) = R(q)x(n)$ et $V(n) = T(q)y(n)$, où q est l'opérateur avance, soient les réponses respectives des deux systèmes lorsqu'ils sont excités par les processus stochastiques stationnaires, centrés, x et y , respectivement. Appelons $\Phi_{xy}(z)$ la transformée en z de la fonction d'intercorrélation des deux processus. Par suite du théorème de Parseval, on a :

$$\langle W(n), V(n) \rangle = \frac{1}{2\pi j} \oint R(z) T^H(z^{-1}) \Phi_{xy}(z) \frac{dz}{z} \quad (2.10)$$

2.2.2 Construction basée sur la représentation d'état équilibrée

L'une des premières généralisations des bases orthogonales est celle dite modèle de Hambo, proposée par [58].

Supposons qu'on veuille construire une base avec l'ensemble de pôles stables¹ $\{\xi_i\}$, $i = 0, 1, \dots, r-1$. La première étape de cette construction consiste à former la fonction de transfert passe-tout prototype $G(z)$ d'ordre r , ayant une représentation équilibrée dans l'espace d'état. Le passe-tout prototype choisi est défini par sa fonction de transfert G .

$$G(z) = \prod_{i=0}^{r-1} \frac{1 - \xi_i^* z}{z - \xi_i} \quad (2.11)$$

$$G(z) = \mathcal{D} + \mathcal{C}(zI - \mathcal{A})^{-1} \mathcal{B} \quad (2.12)$$

¹i.e. pôles à l'intérieur du cercle unité

La transformée en z du vecteur d'état X du système est reliée à celle de l'entrée U par la relation suivante :

$$X(z) = (zI - \mathcal{A})^{-1} \mathcal{B}U(z) = V(z)U(z) \quad (2.13)$$

$V(z)$ est le vecteur qui contient les r fonctions de transfert entrée-état. On peut aussi noter que [97]

$$V(z) = \frac{adj(zI - \mathcal{A})}{\det(zI - \mathcal{A})} \mathcal{B} = \frac{N(z)}{d(z)}, \quad d(z) = \prod_{i=0}^{r-1} (z - \xi_i) \quad (2.14)$$

Le filtre passe-tout prototype G s'écrit alors :

$$G(z) = \frac{z^r d^*(z^{-1})}{d(z)} \quad (2.15)$$

On vérifie aisément que $G^*(z^{-1}) = G^{-1}(z)$.

Proposition 2.1 Les fonctions de transfert définies par $V_i(z) = e_i^T (zI - \mathcal{A})^{-1} \mathcal{B}$, où e_i , $i = 0, 1, \dots, r-1$ désigne le $(i+1)$ -ème vecteur de la base euclidienne $\mathbb{R}^r$, forment un ensemble orthonormal.

Preuve : Supposons que le signal d'entrée u soit blanc, centré, de variance unité. A partir de (2.13) on peut considérer le vecteur d'état comme étant la sortie du filtre SIMO de fonction de transfert $V(z)$ lorsque celui-ci est excité par le signal d'entrée u . Le produit scalaire des signaux d'état est donc donné par l'application directe de la définition (2.10) :

$$\langle x(n), x(n) \rangle = \frac{1}{2\pi j} \oint V(z) V^H(z^{-1}) \frac{dz}{z}$$

Par application du théorème de Parseval, on a :

$$\frac{1}{2\pi j} \oint V(z) V^H(z^{-1}) \frac{dz}{z} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E}\{x(i)x^H(i)\}$$

Or par définition [90], la matrice de contrôlabilité est aussi donnée par

$$\mathcal{P} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E}\{x(i)x^H(i)\}$$

Tenant compte du fait que la représentation d'état du filtre passe-tout prototype G est équilibrée, sa matrice de contrôlabilité est, comme nous l'avons fait remarquer dans les préliminaires, la matrice unité. D'où $\frac{1}{2\pi j} \oint V(z) V^H(z^{-1}) \frac{dz}{z} = I$. ■

Par cette procédure, une base de dimension r , telle que

$$B_{0,i}(z) = V_i(z), \quad i = 0, 1, \dots, r-1$$

2.2. Méthodes de construction des bases de fonctions orthogonales rationnelles

a été générée. Par la multiplication répétée de $V(z)$ par $G(z)$, une base complète de l'espace de Hardy est ensuite générée. Les fonctions de cette base sont données par :

$$B_{k,i}(z) = V_i(z)[G(z)]^k, k = 0, 1, \dots; i = 0, 1, \dots, r-1 \quad (2.16)$$

On vérifie que :

$$\langle B_{k,i}, B_{l,m} \rangle = \frac{1}{2\pi j} \oint V_i(z) G^k(z) V_m^*(z^{-1}) [G^*(z^{-1})]^l \frac{dz}{z} = \delta_{k,l} \delta_{i,m} \quad (2.17)$$

En effet, en notant $B_k(z) = [B_{k,0}(z) \ B_{k,1}(z) \ \dots B_{k,r-1}(z)]^T$, on peut remarquer, en utilisant (2.16) et (2.3), que :

$$B_k(z) B_k^H(z^{-1}) = V(z) V^H(z^{-1}) [G(z) G^*(z^{-1})]^k = V(z) V^H(z^{-1})$$

On a alors :

$$\langle B_k, B_k \rangle = \frac{1}{2\pi j} \oint B_k(z) B_k^H(z^{-1}) \frac{dz}{z} = \frac{1}{2\pi j} \oint V(z) V^H(z^{-1}) \frac{dz}{z} = I$$

Par ailleurs, à partir de (2.14) et de (2.15), on a :

$$\langle B_k, B_l \rangle = \frac{1}{2\pi j} \oint B_k(z) B_l^H(z^{-1}) \frac{dz}{z} = \frac{1}{2\pi j} \oint \chi(z) \frac{dz}{z} = 0$$

avec

$$\chi(z) = z^{r(k-l)} \left(\frac{N(z)}{d(z)} \right) \left(\frac{N^H(z^{-1})}{d^*(z^{-1})} \right) \left(\frac{d(z)}{d^*(z^{-1})} \right)^{l-k}$$

Cette fonction est analytique à l'intérieur du cercle unité pour $l > k$. Par conséquent son intégrale est nulle. D'où : $\langle B_k, B_l \rangle = 0$, $l > k$. Dans le cas où, $l < k$, on sait que $\langle B_k, B_l \rangle = (\langle B_l, B_k \rangle)^H$. Il suffit alors de transconjuguer le résultat précédent pour aboutir à $\langle B_k, B_l \rangle = 0$, $l < k$. En résumé : $\langle B_k, B_l \rangle = \delta_{k,l} I_r$.

La base de fonctions ci-dessus construite est la généralisation des bases orthogonales bien connues comme celle des filtres RIF, de Laguerre et de Kautz.

2.2.2.1 Base de filtres RIF

Considérons la fonction de transfert passe-tout prototype $G(z) = z^{-1}$, ce qui correspond, dans (2.11), à un passe-tout d'ordre $r = 1$ ayant pour pôle $\xi = 0$. Une représentation d'état équilibrée de cette fonction est $(0, 1, 1, 0)$. Par suite, en appliquant (2.13) on obtient : $V(z) = z^{-1}$. La base correspondante est donc :

$$B_k(z) = z^{-(k+1)}$$

C'est la base des filtres RIF pour des systèmes strictement causaux.

2.2.2.2 Base de filtres de Laguerre

Considérons la fonction passe-tout, d'ordre un, $G(z) = \frac{1-\xi z}{z-\xi}$ ayant un pôle réel stable, ξ . Une représentation d'état équilibrée de cette fonction est : $(\xi, \sqrt{1-\xi^2}, \sqrt{1-\xi^2}, -\xi)$. Par application de (2.13) on obtient : $V(z) = \frac{\sqrt{1-\xi^2}}{z-\xi}$. Par suite, la base correspondante est :

$$B_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left(\frac{1-\xi z}{z-\xi} \right)^k$$

C'est la base de filtres de Laguerre.

2.2.2.3 Base de filtres de Kautz

Considérons la fonction de transfert d'un filtre passe-tout d'ordre deux $G(z)$ ayant deux pôles complexes conjugués.

$$G(z) = \frac{-cz^2 + b(c-1)z + 1}{z^2 + b(c-1)z - c}$$

Les constantes b et c sont des réels de valeur absolue strictement inférieure à l'unité², choisies de sorte que $b^2(c-1)^2 + 4c < 0$. Une représentation d'état équilibrée de ce filtre est :

$$\begin{aligned} \mathcal{A} &= \begin{pmatrix} b & \sqrt{1-b^2} \\ c\sqrt{1-b^2} & -bc \end{pmatrix}, \quad \mathcal{B} = \begin{pmatrix} 0 \\ \sqrt{1-c^2} \end{pmatrix} \\ \mathcal{C} &= \begin{pmatrix} \sqrt{(1-c^2)(1-b^2)} & -b\sqrt{1-c^2} \end{pmatrix}, \quad \mathcal{D} = -c \end{aligned}$$

Par application de (2.13), on a :

$$V(z) = \frac{\sqrt{1-c^2}}{z^2 + b(c-1)z - c} \begin{pmatrix} \sqrt{1-b^2} \\ z-b \end{pmatrix}$$

Par suite, la base correspondante est, pour $k = 0, 1, 2, \dots$

$$\begin{aligned} B_{k,0}(z) &= \frac{\sqrt{(1-c^2)(1-b^2)}}{z^2 + b(c-1)z - c} \left[\frac{-cz^2 + b(c-1)z + 1}{z^2 + b(c-1)z - c} \right]^k \\ B_{k,1}(z) &= \frac{\sqrt{1-c^2}(z-b)}{z^2 + b(c-1)z - c} \left[\frac{-cz^2 + b(c-1)z + 1}{z^2 + b(c-1)z - c} \right]^k \end{aligned}$$

C'est la base de filtres de Kautz [133].

²Condition nécessaire pour assurer la stabilité de G

2.2.3 Construction par orthogonalisation de Gram-Schmidt

C'est l'approche dite de Ninness et Gustafsson [90]. Elle consiste en une procédure d'orthonormalisation de Gram-Schmidt appliquée à un ensemble de fonctions $\{\phi_k(z)\}_{k=0}^{\infty}$ pour former une base orthonormale $\{B_k(z)\}_{k=0}^{\infty}$. Selon Gram-Schmidt, la base orthonormale désirée est obtenue en réalisant les étapes suivantes :

$$B_0 = \frac{\phi_0}{\|\phi_0\|} \quad (2.18)$$

Pour $k = 1, 2, \dots$:

$$\beta_k = \phi_k - \sum_{i=0}^{k-1} \langle \phi_k, B_i \rangle B_i \quad (2.19)$$

$$B_k = \frac{\beta_k}{\|\beta_k\|} \quad (2.20)$$

Les fonctions ϕ_k sont choisies de sorte qu'on puisse introduire la connaissance disponible sur les dynamiques du système. Supposons que le système présente des dynamiques proches d'un ensemble de pôles ξ_k , alors les fonctions ϕ_k sont choisies comme :

$$\phi_k(z) = \frac{1}{z - \xi_k} \quad (2.21)$$

Nous considérerons tout d'abord le cas des pôles réels. La première fonction de la base, B_0 , est obtenue en normalisant la fonction ϕ_0 . Cette normalisation est obtenue en appliquant la définition (2.8). On a donc :

$$\|\phi_0\|^2 = \langle \phi_0, \phi_0 \rangle = \frac{1}{2\pi j} \oint \frac{z}{(z - \xi_0)(1 - \xi_0 z)} \frac{dz}{z} = \frac{1}{1 - \xi_0^2}$$

et par suite :

$$B_0(z) = \frac{\sqrt{1 - \xi_0^2}}{z - \xi_0} \quad (2.22)$$

La fonction de base suivante, B_1 , est donnée par la relation de Gram-Schmidt (2.19) :

$$\begin{aligned} \beta_1(z) &= \frac{1}{z - \xi_1} - \left(\frac{1}{2\pi j} \oint \frac{z}{z - \xi_1} \frac{\sqrt{1 - \xi_0^2}}{1 - \xi_0 z} \frac{dz}{z} \right) \frac{\sqrt{1 - \xi_0^2}}{z - \xi_0} \\ &= \frac{1}{z - \xi_1} - \left(\frac{1 - \xi_0^2}{1 - \xi_0 \xi_1} \right) \frac{1}{z - \xi_0} \\ &= \frac{(\xi_1 - \xi_0)(1 - \xi_0 z)}{(1 - \xi_0 \xi_1)(z - \xi_0)(z - \xi_1)} \end{aligned}$$

On vérifie aisément que β_1 admet comme zéro $1/\xi_0$. Par application de la définition (2.8), on a :

$$\|\beta_1\|^2 = \frac{(\xi_0 - \xi_1)^2}{(1 - \xi_0 \xi_1)^2 (1 - \xi_1^2)}$$

et par suite de la normalisation (2.20), on a :

$$B_1(z) = \frac{\sqrt{1-\xi_1^2}}{z-\xi_1} \frac{1-\xi_0 z}{z-\xi_0} \quad (2.23)$$

Supposons à présent que nous avons déjà construit k fonctions de base suivant la procédure de Gram-Schmidt. Alors, la $(k+1)$ -ème fonction sera telle que :

$$\beta_k(z) = \frac{1}{z-\xi_k} - \sum_{i=0}^{k-1} \langle \frac{1}{z-\xi_k}, B_i(z) \rangle B_i(z) = \frac{\Omega_k(z)}{\prod_{i=0}^k (z-\xi_i)}$$

où le numérateur Ω_k est un polynôme en z de degré k .

Proposition 2.2 Les valeurs $\{1/\xi_i\}$, $i = 0, 1, \dots, k-1$ sont les zéros du polynôme $\Omega_k(z)$.

Preuve : Par construction on sait que $0 = \langle \beta_k, B_i \rangle = (\langle B_i, \beta_k \rangle)^*$, $\forall i < k$. Evaluons alors, pour chacune de ces valeurs de i , le produit scalaire ci-dessus :

$$\begin{aligned} \langle B_0, \beta_k \rangle &= \frac{1}{2\pi j} \oint B_0(z) \beta_k^*(z^{-1}) \frac{dz}{z} = \sqrt{1-\xi_0^2} \beta_k^*(1/\xi_0) \\ \langle \beta_k, B_0 \rangle &= (\langle B_0, \beta_k \rangle)^* = \sqrt{1-\xi_0^2} \beta_k(1/\xi_0) \\ 0 &= \langle \beta_k, B_0 \rangle = \sqrt{1-\xi_0^2} \beta_k(1/\xi_0) \end{aligned}$$

$1/\xi_0$ est donc un zéro de $\Omega_k(z)$. On en déduit la factorisation $\Omega_k(z) = (1-\xi_0 z) \Omega_{k-1}(z)$. Calculons ensuite le produit $\langle B_1, \beta_k \rangle$ en utilisant la factorisation de $\Omega_k(z)$ ci-dessus :

$$\begin{aligned} \langle B_1, \beta_k \rangle &= \frac{1}{2\pi j} \oint \frac{\sqrt{1-\xi_1^2} (1-\xi_0 z)}{(z-\xi_1)(z-\xi_0)} \frac{(1-\xi_0 z^{-1}) \Omega_{k-1}^*(z^{-1})}{\prod_{i=0}^k (z^{-1}-\xi_i)} \frac{dz}{z} \\ &= \frac{1}{2\pi j} \oint \frac{\sqrt{1-\xi_1^2}}{z-\xi_1} \frac{\Omega_{k-1}^*(z^{-1})}{\prod_{i=1}^k (z^{-1}-\xi_i)} \frac{dz}{z} \\ &= \frac{\sqrt{1-\xi_1^2} \Omega_{k-1}^*(1/\xi_1)}{\prod_{i=1}^k (\xi_1^{-1}-\xi_i)} \end{aligned}$$

Suivant la même procédure, on montre que $1/\xi_1$ est un zéro de Ω_{k-1} et donc de Ω_k . En itérant ce processus pour toutes les fonctions déjà construites, on trouve tous les zéros de $\Omega(z)$ sous la forme $1/\xi_i$, $i < k$. ■

2.2. Méthodes de construction des bases de fonctions orthogonales rationnelles

A une constante près A , le polynôme $\Omega_k(z)$ s'écrit $\Omega_k(z) = A \prod_{i=0}^{k-1} (1 - \xi_i z)$. Par suite, la normalisation (2.20) implique :

$$B_k(z) = \frac{\sqrt{1 - \xi_k^2}}{z - \xi_k} \prod_{i=0}^{k-1} \frac{1 - \xi_i z}{z - \xi_i} \quad (2.24)$$

Pour des pôles réels, la base définie par les relations (2.22) et (2.24) est aussi appelée base de Laguerre généralisée.

Cette procédure de construction de la BFOR reste valable dans le cas où les pôles sont complexes. Auquel cas nous avons :

$$B_k(z) = \frac{\sqrt{1 - |\xi_k|^2}}{z - \xi_k} \prod_{i=0}^{k-1} \frac{1 - \xi_i^* z}{z - \xi_i} \quad (2.25)$$

Cependant, les réponses impulsionnelles de ces filtres sont complexes. C'est aussi le cas lorsque les k premiers pôles sont réels et qu'un $(k+1)$ -ème pôle complexe est introduit. En conséquence, le système résultant est physiquement non réalisable. Pour le rendre réalisable, la solution consiste à introduire les modes complexes par paire conjuguée.

Pour introduire un mode complexe, on détermine selon (2.24) les fonctions de base B_k et B_{k+1} correspondant à la paire de pôles complexes conjugués choisis, c'est-à-dire $\xi_{k+1} = \xi_k^*$. A partir de ces fonctions, on génère une nouvelle paire de fonctions de base ayant respectivement une réponse impulsionnelle réelle. Ces deux nouvelles fonctions sont des combinaisons linéaires des précédentes, ce qui permet de préserver l'orthogonalité de la base. Une fois déterminées, elles remplaceront B_k et B_{k+1} dans la base précédemment construite.

$$\begin{pmatrix} B'_k \\ B''_k \end{pmatrix} = \begin{pmatrix} c_0 & c_1 \\ c'_0 & c'_1 \end{pmatrix} \begin{pmatrix} B_k \\ B_{k+1} \end{pmatrix} \quad (2.26)$$

Pour les besoins de la suite de ce développement, déterminons les normes respectives de ces nouvelles fonctions ainsi que leur produit scalaire. On a :

$$\begin{aligned} \|B'_k\|^2 &= \langle c_0 B_k + c_1 B_{k+1}, c_0 B_k + c_1 B_{k+1} \rangle \\ &= |c_0|^2 \langle B_k, B_k \rangle + |c_1|^2 \langle B_{k+1}, B_{k+1} \rangle \\ &\quad + c_0 c_1^* \langle B_k, B_{k+1} \rangle + c_1 c_0^* \langle B_{k+1}, B_k \rangle \\ &= |c_0|^2 + |c_1|^2 \end{aligned} \quad (2.27)$$

$$\begin{aligned} \|B''_k\|^2 &= \langle c'_0 B_k + c'_1 B_{k+1}, c'_0 B_k + c'_1 B_{k+1} \rangle \\ &= |c'_0|^2 + |c'_1|^2 \end{aligned} \quad (2.28)$$

$$a = \langle B'_k, B''_k \rangle = c_0 c'^*_0 + c_1 c'^*_1 \quad (2.29)$$

Nous pouvons souligner que les fonctions B'_k et B''_k engendrent un plan orthogonal au sous-espace engendré par $\{B_i(z)\}_{i=0}^{k-1}$. Cependant, il y a une infinité de choix possibles car sur un plan vectoriel le nombre de paires de vecteurs orthonormés est infini.

La forme récursive de la relation (2.25) est donnée par :

$$B_k(z) = \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{1-\xi_{k-1}^* z}{z-\xi_k} \right) B_{k-1}(z) \quad (2.30)$$

Tenant compte du fait que $\xi_{k+1} = \xi_k^*$, la relation de récurrence ci-dessus permet d'écrire

$$B_{k+1}(z) = \frac{1-\xi_k^* z}{z-\xi_k^*} B_k(z)$$

Par suite

$$B_{k+1}(z) = \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{(1-\xi_{k-1}^* z)(1-\xi_k^* z)}{z^2 - (\xi_k + \xi_k^*)z + |\xi_k|^2} \right) B_{k-1}(z) \quad (2.31)$$

Tenant compte des équations (2.26), (2.30) et (2.31), nous avons :

$$\begin{aligned} B'_k(z) &= \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{(1-\xi_{k-1}^* z)((c_0 - c_1 \xi_k^*)z + c_1 - c_0 \xi_k^*)}{z^2 - (\xi_k + \xi_k^*)z + |\xi_k|^2} \right) B_{k-1}(z) \\ B''_k(z) &= \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{(1-\xi_{k-1}^* z)((c'_0 - c'_1 \xi_k^*)z + c'_1 - c'_0 \xi_k^*)}{z^2 - (\xi_k + \xi_k^*)z + |\xi_k|^2} \right) B_{k-1}(z) \end{aligned}$$

En posant :

$$\alpha = c_0 - c_1 \xi_k^* \quad \mu = c_1 - c_0 \xi_k^*$$

$$\alpha' = c'_0 - c'_1 \xi_k^* \quad \mu' = c'_1 - c'_0 \xi_k^*$$

nous avons

$$\begin{aligned} c_0 &= \frac{\alpha + \xi_k^* \mu}{1 - \xi_k^{*2}} & c_1 &= \frac{\alpha \xi_k^* + \mu}{1 - \xi_k^{*2}} \\ c'_0 &= \frac{\alpha' + \xi_k^* \mu'}{1 - \xi_k^{*2}} & c'_1 &= \frac{\alpha' \xi_k^* + \mu'}{1 - \xi_k^{*2}} \end{aligned} \quad (2.32)$$

Par suite :

$$\begin{aligned} B'_k(z) &= \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{(1-\xi_{k-1}^* z)(\alpha z + \mu)}{z^2 - (\xi_k + \xi_k^*)z + |\xi_k|^2} \right) B_{k-1}(z) \\ B''_k(z) &= \frac{\sqrt{1-|\xi_k|^2}}{\sqrt{1-|\xi_{k-1}|^2}} \left(\frac{(1-\xi_{k-1}^* z)(\alpha' z + \mu')}{z^2 - (\xi_k + \xi_k^*)z + |\xi_k|^2} \right) B_{k-1}(z) \end{aligned}$$

Pour assurer la normalité de ces deux fonctions de base, les relations (2.27) et (2.28) engendrent alors les contraintes $|c_0|^2 + |c_1|^2 = 1$ et $|c'_0|^2 + |c'_1|^2 = 1$, qui s'écrivent aussi, à partir de (2.32) et en supposant réels les coefficients $\alpha, \alpha', \mu, \mu'$ [90] :

$$\begin{aligned} (1 + |\xi_k|^2)(\alpha^2 + \mu^2) + 2(\xi_k + \xi_k^*)\alpha\mu &= |1 - \xi_k^2|^2 \\ (1 + |\xi_k|^2)(\alpha'^2 + \mu'^2) + 2(\xi_k + \xi_k^*)\alpha'\mu' &= |1 - \xi_k^2|^2 \end{aligned}$$

2.2. Méthodes de construction des bases de fonctions orthogonales rationnelles

En posant $w_1 = (\alpha \quad \mu)^T$ et $w_2 = (\alpha' \quad \mu')^T$, les relations ci-dessus s'écrivent de façon plus compacte telles que

$$w_1^T \mathbf{M} w_1 = |1 - \xi_k^2|^2 \quad w_2^T \mathbf{M} w_2 = |1 - \xi_k^2|^2 \quad (2.33)$$

avec

$$\mathbf{M} = \begin{pmatrix} 1 + |\xi_k|^2 & 2\operatorname{Re}(\xi_k) \\ 2\operatorname{Re}(\xi_k) & 1 + |\xi_k|^2 \end{pmatrix}$$

Pour assurer l'orthogonalité des deux nouvelles fonctions de base, (2.29) doit s'annuler. Par suite, on obtient une nouvelle équation quadratique :

$$w_1^T \mathbf{M} w_2 = 0 \quad (2.34)$$

On a donc trois équations pour déterminer les quatre variables α, α', μ et μ' . En conséquence, plusieurs vecteurs w_1 et w_2 solutions des équations (2.33) et (2.34) existent. Un choix possible est [82] :

$$w_1 = \sqrt{\frac{(1 + \xi_k^*)(1 + \xi_k)}{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad w_2 = \sqrt{\frac{(1 - \xi_k^*)(1 - \xi_k)}{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

On obtient alors :

$$B'_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{2}} \frac{z - 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \prod_{i=0}^{k-1} \frac{1 - \xi_i z}{z - \xi_i}$$

$$B''_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{2}} \frac{z + 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \prod_{i=0}^{k-1} \frac{1 - \xi_i z}{z - \xi_i}$$

Considérons à présent le cas où tous les pôles sont complexes. Dans ce cas, on a la formulation suivante :

$$B_{2k}(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{2}} \frac{z - 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \prod_{i=0}^{k-1} \frac{(1 - \xi_i^* z)(1 - \xi_i z)}{(z - \xi_i)(z - \xi_i^*)} \quad (2.35)$$

$$B_{2k+1}(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{2}} \frac{z + 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \prod_{i=0}^{k-1} \frac{(1 - \xi_i^* z)(1 - \xi_i z)}{(z - \xi_i)(z - \xi_i^*)} \quad (2.36)$$

La base définie par (2.35) et (2.36) peut-être appelée Base de Kautz généralisée.

Par ailleurs, [90] font remarquer que les nombreuses possibilités de choix des solutions des équations (2.33) et (2.34) permettent de générer des bases, pour des dynamiques de second ordre, autres que celle de Kautz.

La base de fonctions construite par la méthode de Ninness et Gustafsson est la généralisation des bases orthogonales bien connues comme celle des filtres RIF, de Laguerre et de Kautz.

2.2.3.1 Base de filtres RIF

La procédure de construction est réalisée avec la fonction $\phi(z) = z^{-1}$. A partir de la formulation générale (2.24), il suffit de prendre $\xi_k = 0, \forall k$. D'où :

$$B_k(z) = z^{-(k+1)}$$

C'est la base des filtres RIF pour des systèmes strictement causaux.

2.2.3.2 Base de filtres de Laguerre

La construction de la base est réalisée avec $\phi_k(z) = \frac{1}{z-\xi}, \forall k$. Autrement dit, à partir de la formulation générale, il suffit de prendre $\xi_k = \xi, \forall k$. D'où :

$$B_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left(\frac{1-\xi z}{z-\xi} \right)^k$$

C'est la base de filtres de Laguerre.

2.2.3.3 Base de filtres de Kautz

On considère ici l'introduction répétée d'un seul mode complexe ξ . Les formules (2.35) et (2.36) donnent alors :

$$\begin{aligned} B_{2k}(z) &= A(\xi) \frac{z-1}{z^2 - 2\operatorname{Re}(\xi)z + |\xi|^2} \left(\frac{(1-\xi^*z)(1-\xi z)}{z^2 - 2\operatorname{Re}(\xi)z + |\xi|^2} \right)^k \\ B_{2k+1}(z) &= A'(\xi) \frac{z+1}{z^2 - 2\operatorname{Re}(\xi)z + |\xi|^2} \left(\frac{(1-\xi^*z)(1-\xi z)}{z^2 - 2\operatorname{Re}(\xi)z + |\xi|^2} \right)^k \end{aligned}$$

avec

$$\begin{aligned} A(\xi) &= \sqrt{\frac{(1-|\xi|^2)(1+\xi)(1+\xi^*)}{2}} \\ A'(\xi) &= \sqrt{\frac{(1-|\xi|^2)(1-\xi)(1-\xi^*)}{2}} \end{aligned}$$

On retrouve la base de Kautz telle que définie par [96]. C'est la seconde définition des fonctions de Kautz. Elle engendre le même espace que celles définies dans le paragraphe 2.2.2. On montre que ces deux définitions sont équivalentes [82]. On passe de l'une vers l'autre via une rotation. Notons que par un autre choix de solutions des équations (2.33) et (2.34) on retrouve directement la première définition des fonctions de Kautz données en 2.2.2 [90].

2.2.4 Méthode générale de construction

Le modèle de Hambo, défini dans le paragraphe 2.2.2 peut aussi être vu comme résultant d'une mise en cascade d'un même filtre passe-tout G d'ordre quelconque. Quant à la construction de Ninness et Gustafsson, on peut remarquer que les fonctions de transfert de type $\frac{1-\xi_i^* z}{z-\xi_i}$ que l'on retrouve dans la formule (2.25) et celles de type $\frac{(1-\xi_i^* z)(1-\xi_i z)}{(z-\xi_i)(z-\xi_i^*)}$ dans (2.35) et (2.36) sont respectivement de type passe-tout du premier et du second ordre. Par conséquent, les fonctions de la BFOR ainsi construite peuvent être vues comme résultant de la mise en cascade de différents filtres passe-tout mais de même ordre. Une méthode de construction générale consiste donc à mettre en cascade différents filtres passe-tout d'ordre quelconque [15, 13, 31].

Pour ce faire, considérons l'ensemble de filtres passe-tout $\{G_k(z)\}_{k \in \mathbb{N}}$ ayant respectivement r_k pôles stables. Ils sont définis par :

$$G_k(z) = \prod_{j=1}^{r_k} \frac{1 - \xi_{k,j}^*}{z - \xi_{k,j}} \quad (2.37)$$

Pour la construction de la base, on utilise les représentations d'état $(\mathcal{A}_k, \mathcal{B}_k, \mathcal{C}_k, \mathcal{D}_k)$ des filtres passe-tout G_k ; nous prendrons des représentations équilibrées. Les fonctions de transfert entrée-état de chaque filtre étant :

$$V_k(z) = (zI - \mathcal{A}_k)^{-1} \mathcal{B}_k$$

on construit les fonctions de base de la manière suivante :

$$B_k(z) = V_k(z) \prod_{i=0}^{k-1} G_i(z) \quad (2.38)$$

Comme pour le modèle de Hambo, on peut aussi se ramener au calcul des résidus pour montrer l'orthonormalité de la base ainsi construite. Cependant, [15] en utilisant les résultats de [117] ont mis en lumière des propriétés intéressantes, utiles pour prouver l'orthonormalité. En effet, le principal résultat de [117] est que l'orthogonalité et les propriétés des passe-tout sont préservées pour plusieurs connexions différentes de ces filtres. Par exemple pour une connexion en cascade, le résultat suivant a été prouvé dans [117].

Théorème 2.1 Soient $H_1(z)$ et $H_2(z)$ deux filtres passe-tout orthogonaux ayant respectivement les états x_1 et x_2 . Alors le filtre résultant de leur mise en cascade, $H_1 H_2$ avec les états $(x_1^T \ x_2^T)^T$ est aussi un filtre passe-tout orthogonal.

Preuve : Considérons H_1 et H_2 ayant les représentations d'état respectives $(\mathcal{A}_1, \mathcal{B}_1, \mathcal{C}_1, \mathcal{D}_1)$ et $(\mathcal{A}_2, \mathcal{B}_2, \mathcal{C}_2, \mathcal{D}_2)$. Leur mise en cascade est illustrée par la figure 2.1. Relativement à la relation (2.6), on note :

$$\mathbf{F}_1 = \begin{pmatrix} \mathcal{A}_1 & \mathcal{B}_1 \\ \mathcal{C}_1 & \mathcal{D}_1 \end{pmatrix} \quad \mathbf{F}_2 = \begin{pmatrix} \mathcal{A}_2 & \mathcal{B}_2 \\ \mathcal{C}_2 & \mathcal{D}_2 \end{pmatrix}$$

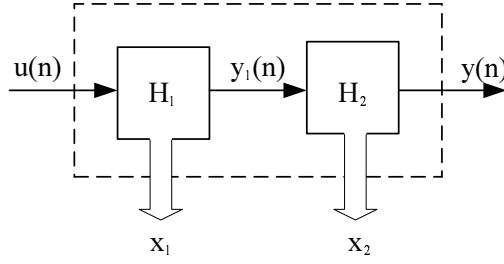


FIG. 2.1 – Mise en cascade de deux passe-tout

Déterminons à présent la représentation d'état du système résultant. On a :

$$\begin{aligned} x_1(n+1) &= \mathcal{A}_1 x_1(n) + \mathcal{B}_1 u(n) \\ y_1(n) &= \mathcal{C}_1 x_1(n) + \mathcal{D}_1 u(n) \\ x_2(n+1) &= \mathcal{A}_2 x_2(n) + \mathcal{B}_2 y_1(n) = \mathcal{A}_2 x_2(n) + \mathcal{B}_2 \mathcal{C}_1 x_1(n) + \mathcal{B}_2 \mathcal{D}_1 u(n) \\ y(n) &= \mathcal{C}_2 x_2(n) + \mathcal{D}_2 y_1(n) = \mathcal{C}_2 x_2(n) + \mathcal{D}_2 \mathcal{C}_1 x_1(n) + \mathcal{D}_2 \mathcal{D}_1 u(n) \end{aligned}$$

Le filtre résultant a pour vecteur d'état $(x_1^T \ x_2^T)^T$. On lui associe donc la matrice partitionnée $\mathbf{F}_3$ ci-dessous.

$$\mathbf{F}_3 = \left(\begin{array}{cc|c} \mathcal{A}_1 & 0 & \mathcal{B}_1 \\ \mathcal{B}_2 \mathcal{C}_1 & \mathcal{A}_2 & \mathcal{B}_2 \mathcal{D}_1 \\ \hline \mathcal{D}_2 \mathcal{C}_1 & \mathcal{C}_2 & \mathcal{D}_2 \mathcal{D}_1 \end{array} \right)$$

Cette matrice est ensuite factorisée en un produit de deux matrices de sorte que :

$$\mathbf{F}_3 = \begin{pmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathcal{A}_2 & \mathcal{B}_2 \\ 0 & \mathcal{C}_2 & \mathcal{D}_2 \end{pmatrix} \begin{pmatrix} \mathcal{A}_1 & 0 & \mathcal{B}_1 \\ 0 & \mathbf{I} & 0 \\ \mathcal{C}_1 & 0 & \mathcal{D}_1 \end{pmatrix}$$

Tenant compte de l'orthogonalité respective des matrices $\mathbf{F}_1$ et $\mathbf{F}_2$, on peut aisément vérifier celle des deux matrices en facteur et par suite, celle de leur produit. ■

De ce théorème, il est évident de constater qu'à partir d'une famille de filtres passe-tout stables G_k , d'ordre respectif r_k , une famille orthonormale est générée en mettant en cascade ces filtres. Les fonctions de base sont alors les fonctions de transfert entrée-états de chaque passe-tout.

[90] et [15] ont montré qu'une condition nécessaire et suffisante pour que cette base soit complète dans l'espace de Hardy est :

$$\sum_{k=0}^{\infty} (1 - \xi_k) = \infty \quad (2.39)$$

Cette méthode de construction généralise bel et bien la construction de Heuberger et al. ainsi que celle de Ninness et Gustafsson, et par suite celle des bases de filtres RIF, de La-guerre et de Kautz. Ainsi, de cette construction, on obtient le modèle de Hambo en met-tant le même filtre passe-tout d'ordre q quelconque en cascade. Quant à la base de Nin-ness et Gustafsson, elle est obtenue en mettant en cascade différents filtres passe-tout de premier ordre, de représentation équilibrée respective $(\xi_k, \sqrt{1-|\xi_k|^2}, \sqrt{1-|\xi_k|^2}, -\xi_k^*)$. Toutefois, une telle représentation génère une réponse impulsionnelle complexe dans le cas où au moins un pôle est complexe. Le système physique résultant est non réali-sable. Il faut donc tenir compte de la procédure d'insertion de modes complexes telle que décrite dans le paragraphe 2.2.3. La base sera donc obtenue en mettant en cas-cade des passe-tout du premier ordre $G_k(z) = \frac{1-\xi_k z}{z-\xi_k}$, de représentation d'état respec-tive $(\xi_k, \sqrt{1-|\xi_k|^2}, \sqrt{1-|\xi_k|^2}, -\xi_k^*)$, pour des pôles réels et des passe-tout du second ordre $G_k(z) = \frac{(1-\xi_k z)(1-\xi_k^* z)}{(z-\xi_k)(z-\xi_k^*)}$, de représentation d'état respective

$$\mathcal{A}_k = \begin{pmatrix} b_k & \sqrt{1-b_k^2} \\ c_k \sqrt{1-b_k^2} & -b_k c_k \end{pmatrix}, \quad \mathcal{B}_k = \begin{pmatrix} 0 \\ \sqrt{1-c_k^2} \end{pmatrix}$$

$$\mathcal{C}_k = \begin{pmatrix} \sqrt{(1-c_k^2)(1-b_k^2)} & -b_k \sqrt{1-c_k^2} \end{pmatrix}, \quad \mathcal{D}_k = -c_k$$

avec $c_k = -|\xi_k|^2$ et $b_k = \frac{2\text{Re}(\xi_k)}{1+|\xi_k|^2}$.

2.3 Réseaux de filtres associés à la base de fonctions or-thogonales rationnelles

La base de fonctions orthogonales rationnelles construite par les méthodes sus-indiquées étant complète, tout système stable et causal H peut-être développé sur celle-ci :

$$H(z) = \sum_{k=0}^{\infty} g_k B_k(z) \quad (2.40)$$

Les g_k sont appelés coefficients de Fourier. Ils sont donnés par :

$$g_k = \langle H(z), B_k(z) \rangle = \sum_{i=0}^{\infty} h(i) b_k(i) \quad (2.41)$$

où $h(i) = \mathcal{Z}^{-1} \{H(z)\}$ et $b_k(i) = \mathcal{Z}^{-1} \{B_k(z)\}$.

La sortie $y(n)$ du système H correspondant à une entrée $u(n)$ est telle que

$$y(n) = H(z)u(n) = \sum_{k=0}^{\infty} g_k B_k(z)u(n) = \sum_{k=0}^{\infty} g_k s_k(n)$$

où q désigne l'opérateur avance.

Comme pour les filtres RIF, des structures d'implémentation transversale ou orthogonale (par exemple en treillis) peuvent être définies.

2.3.1 Structure transversale

Pour le modèle de [90], nous pouvons développer deux types d'implémentation transversale. La première est celle présentée par les auteurs (figure 2.2).

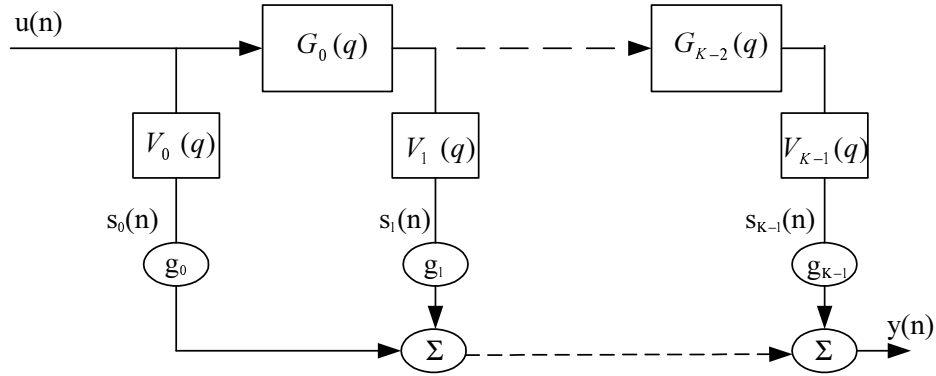


FIG. 2.2 – Réseau de filtres de la BFOR de Ninness et Gustafsson

Les fonctions de base sont données par $B_k(z) = V_k(z) \prod_{i=0}^{k-1} G_i(z)$ avec :

– cas des pôles réels

$$V_k(z) = \frac{\sqrt{1 - \xi_k^2}}{z - \xi_k} \quad G_i(z) = \frac{1 - \xi_i z}{z - \xi_i}$$

– cas des modes complexes

$$V_k(z) = \begin{pmatrix} \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{2}} \frac{z - 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \\ \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{2}} \frac{z + 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2} \end{pmatrix} \quad G_i(z) = \frac{(1 - \xi_i z)(1 - \xi_i^* z)}{(z - \xi_i)(z - \xi_i^*)}$$

Il est à noter que sur la figure 2.2, les $s_k(n)$ et les g_k sont scalaires si G_k est du premier ordre et vectoriels de dimension 2 si G_k est du second ordre.

La seconde forme d'implémentation tient compte des relations de récurrence entre deux fonctions de base consécutives. Elle a été développée par [82]. On peut cependant remarquer que cette représentation n'est valable que pour le cas des pôles réels. Dans le

cas où la base est formée à partir de pôles complexes et réels, nous développons différentes relations de récurrence suivant l'ordre de placement des pôles conduisant à la structure en cascade de la figure 2.3. Pour $k = 0$, on a :

– ξ_0 est réel :

$$B_0(z) = \frac{\sqrt{1 - \xi_0^2}}{z - \xi_0}$$

– ξ_0 est complexe :

$$B'_0(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{2}} \frac{z - 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

$$B''_0(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{2}} \frac{z + 1}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

Pour $k > 0$, on a :

– ξ_k et ξ_{k-1} sont réels :

$$B_k(z) = L_k(z)B_{k-1}(z)$$

$$L_k(z) = \frac{\sqrt{1 - \xi_k^2}}{\sqrt{1 - \xi_{k-1}^2}} \left(\frac{1 - \xi_{k-1}z}{z - \xi_k} \right)$$

– ξ_k est complexe et ξ_{k-1} réel :

$$B'_k(z) = L'_k(z)B_{k-1}(z)$$

$$L'_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{2(1 - \xi_{k-1}^2)}} \frac{(z - 1)(1 - \xi_{k-1}z)}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

$$B''_k(z) = L''_k(z)B_{k-1}(z)$$

$$L''_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{2(1 - \xi_{k-1}^2)}} \frac{(z + 1)(1 - \xi_{k-1}z)}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

– ξ_k et ξ_{k-1} sont complexes :

$$B'_k(z) = L'_k(z)B'_{k-1}(z)$$

$$L'_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 + \xi_k)(1 + \xi_k^*)}{(1 - |\xi_{k-1}|^2)(1 + \xi_{k-1})(1 + \xi_{k-1}^*)}} \frac{(1 - \xi_{k-1}^*z)(1 - \xi_{k-1}z)}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

$$B''_k(z) = L''_k(z)B''_{k-1}(z)$$

$$L''_k(z) = \sqrt{\frac{(1 - |\xi_k|^2)(1 - \xi_k)(1 - \xi_k^*)}{(1 - |\xi_{k-1}|^2)(1 - \xi_{k-1})(1 - \xi_{k-1}^*)}} \frac{(1 - \xi_{k-1}^*z)(1 - \xi_{k-1}z)}{z^2 - 2\operatorname{Re}(\xi_k)z + |\xi_k|^2}$$

- ξ_k est réel et ξ_{k-1} complexe. Nous éviterons ce cas, car il fait intervenir un filtre non stable. En effet

$$B_k(z) = L'_k(z)B'_{k-1}(z)$$

$$L'_k(z) = \sqrt{\frac{2(1-|\xi_k|^2)}{(1-|\xi_{k-1}|^2)(1+\xi_{k-1})(1+\xi_{k-1}^*)}} \frac{(1-\xi_{k-1}z)(1-\xi_{k-1}^*z)}{(z-\xi_k)(z-1)}$$

ou

$$B_k(z) = L''_k(z)B''_{k-1}(z)$$

$$L''_k(z) = \sqrt{\frac{2(1-|\xi_k|^2)}{(1-|\xi_{k-1}|^2)(1-\xi_{k-1})(1-\xi_{k-1}^*)}} \frac{(1-\xi_{k-1}z)(1-\xi_{k-1}^*z)}{(z-\xi_k)(z+1)}$$

Dans le cas où tous les pôles sont réels on a la réalisation suivante :

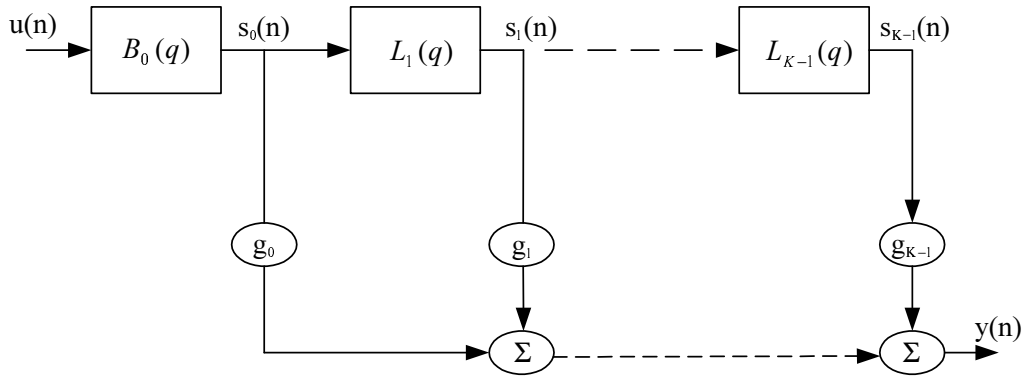


FIG. 2.3 – Réseau de filtres de la BFOR de Ninness et Gustafsson : cas des pôles réels

2.3.2 Structure orthogonale : Filtre en treillis

Dans la littérature, on retrouve quelques références sur l'implémentation des bases orthogonales en utilisant une structure de filtres en treillis. Ces filtres assurent, en plus de l'orthogonalité de la base, celle des entrées filtrées. Par ailleurs, il est bien connu que cette structure de filtre présente des avantages certains quant à la stabilité numérique des algorithmes. [95] et [88] ont développé un modèle de Laguerre en treillis tandis que [97] présente un modèle de Hambo en treillis. Ce modèle généralise à la fois l'implémentation en treillis des filtres FIR et des filtres de Laguerre. Quoique nous n'ayons trouvé aucune référence, dans la littérature, sur l'implémentation en treillis des filtres

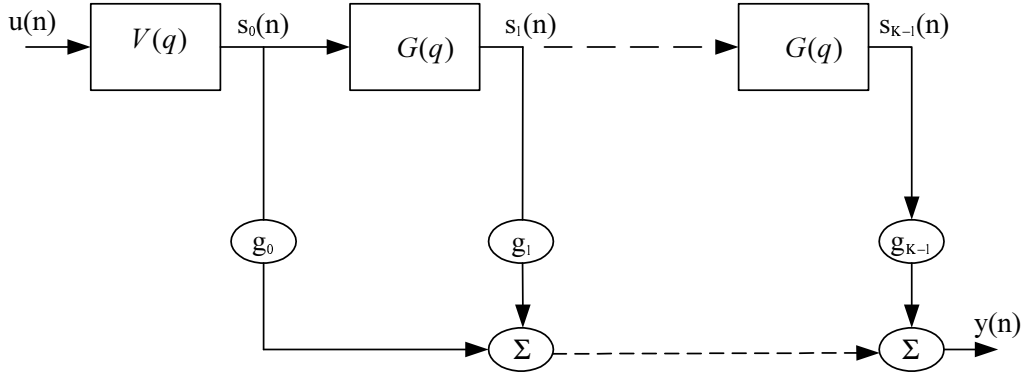


FIG. 2.4 – Modèle de Hambo ou réseau de filtres de la BFOR de Heuberger *et al.*

de Kautz, il est possible d'en dériver une à partir de la formulation générale du modèle de Hambo en treillis [97]. Il suffira pour ce faire d'utiliser un passe-tout du second ordre.

Rappelons que les fonctions de base du modèle de Hambo sont telles que $B_k(z) = V(z)G^k(z)$. G étant la fonction de transfert scalaire d'un filtre passe-tout orthogonal d'ordre r , $V(z)$ est dans ce cas un vecteur de dimension r . Dans la représentation de la figure 2.4, les $s_k(n)$ et les g_k sont donc des vecteurs de dimension r .

Comme on peut le remarquer sur la figure 2.4, les entrées filtrées du système sont :

$$s_k(n) = B_k(q)u(n) = V(q)G^k(q)u(n)$$

Supposons que u est un processus stochastique stationnaire. De la propriété des passe-tout orthogonaux (2.3), on a : $G^*(z^{-1}) = G^{-1}(z)$. Conformément à la définition (2.9) nous pouvons alors écrire :

$$\langle s_k, s_m \rangle = G^{k-m}(z)V_0(z)V_0^H(z^{-1})\Gamma_u \quad (2.42)$$

où $\Gamma_u = \mathbb{E}[uu^*]$. Ce produit scalaire ne dépend que de la différence entre les indices k et m . Par conséquent, la matrice de corrélation des entrées filtrées est une matrice de Toeplitz, de dimension $r \times r$. Grâce à cette structure Toeplitz, un filtre en treillis peut être déterminé [57].

Il est à noter que dans le cas de la construction de [90] la matrice de corrélation des entrées filtrées n'est pas de type Toeplitz. En effet, comme on peut le remarquer à partir de la figure 2.2, les entrées filtrées du système sont :

$$s_k(n) = B_k(q)u(n) = V_k(q) \prod_{i=0}^{k-1} G_i(q)u(n)$$

En supposant que u est un processus stochastique stationnaire, par suite du théorème de Parseval on a :

$$\langle s_k, s_l \rangle = \frac{1}{2\pi j} \oint V_k(z)V_l^*(z^{-1}) \prod_{i=0}^{k-1} G_i(z) \prod_{m=0}^{l-1} G_m^*(z^{-1}) \Phi_{uu}(z) \frac{dz}{z}$$

Sachant que $G_i^*(z^{-1}) = G_i^{-1}(z)$, $\forall i$, on a les trois cas suivants :

$$\begin{aligned} \langle s_k, s_l \rangle &= \frac{1}{2\pi j} \oint V_k(z) V_l^*(z^{-1}) \prod_{i=l}^{k-1} G_i(z) \Phi_{uu}(z) \frac{dz}{z}, \quad \text{si } k > l \\ \langle s_k, s_l \rangle &= \frac{1}{2\pi j} \oint V_k(z) V_l^*(z^{-1}) \prod_{i=k}^{l-1} G_i^{-1}(z) \Phi_{uu}(z) \frac{dz}{z}, \quad \text{si } k < l \\ \langle s_k, s_k \rangle &= \frac{1}{2\pi j} \oint V_k(z) V_k^*(z^{-1}) \Phi_{uu}(z) \frac{dz}{z} \end{aligned}$$

On constate que ce produit scalaire, qui définit la matrice de corrélation des entrées filtrées, ne dépend pas du décalage $k - l$. La structure de la matrice de corrélation des entrées filtrées n'est donc pas, dans ce cas, de type Toeplitz. D'où l'impossibilité de dériver une implémentation en treillis de la BFOR de Ninness et Gustafsson. Toutefois, l'orthogonalisation des entrées filtrées par la BFOR peut être réalisée en introduisant un module d'orthogonalisation à la suite du filtrage par la BFOR. Nous reviendrons sur la détermination de ce module d'orthogonalisation dans le chapitre 4.

2.4 Choix de la base optimale

Dans cette section, nous considérons la BFOR de Ninness et Gustafsson [90]. Dans ce cas, les signaux d'entrée filtrée s_k sont scalaires.

2.4.1 Position du problème

Tout système stable et causal H admet un développement infini sur une BFOR $\mathfrak{B} = \{B_k\}_{k=0}^{\infty}$. Pour des raisons pratiques évidentes, nous considérons H_K son développement tronqué à un ordre fini K . On a : $H(z) = \sum_{k=0}^{\infty} g_k B_k(z)$ et $H_K(z) = \sum_{k=0}^{K-1} g_k B_k(z)$. Notons $u(n)$, $d(n)$ et $y(n)$ respectivement le signal d'entrée du système, la sortie du système et la sortie du modèle. Cette dernière s'écrit :

$$y(n) = \sum_{i=0}^{\infty} h(i) u(n-i) = \sum_{i=0}^{\infty} \sum_{k=0}^{K-1} g_k b_k(i) u(n-i) = \sum_{k=0}^{K-1} g_k s_k(n) = G^T S(n)$$

avec :

$$\begin{aligned} s_k(n) &= \sum_{i=0}^{\infty} b_k(i) u(n-i) \\ G &= (g_0 \quad g_1 \quad \cdots \quad g_{K-1})^T \\ S(n) &= (s_0(n) \quad s_1(n) \quad \cdots \quad s_{K-1}(n))^T \end{aligned}$$

2.4. Choix de la base optimale

Rappelons que le vecteur optimal des coefficients de Fourier au sens de l'erreur quadratique moyenne minimale (EQMM) est :

$$\hat{G} = \Gamma_S^{-1} \mathfrak{C}_{dS} \quad (2.43)$$

avec

$$\Gamma_S = \mathbb{E} [S(n)S^T(n)] \quad \mathfrak{C}_{dS} = \mathbb{E} [d(n)S(n)]$$

La meilleure approximation du système au sens de l'EQMM est alors :

$$H_K(z) = \sum_{k=0}^{K-1} \hat{g}_k B_k(z)$$

Evaluons l'erreur quadratique d'approximation du système :

$$\begin{aligned} \|H(z) - H_K(z)\|^2 &= \left\langle \sum_{j=0}^{\infty} g_j B_j(z) - \sum_{k=0}^{K-1} \hat{g}_k B_k(z), \sum_{j=0}^{\infty} g_j B_j(z) - \sum_{k=0}^{K-1} \hat{g}_k B_k(z) \right\rangle \\ &= \sum_{j=0}^{\infty} g_j^2 - \sum_{j=0}^{\infty} \sum_{k=0}^{K-1} g_j \hat{g}_k \langle B_j(z), B_k(z) \rangle \\ &\quad - \sum_{k=0}^{K-1} \sum_{j=0}^{\infty} \hat{g}_k g_j \langle B_k(z), B_j(z) \rangle + \sum_{k=0}^{K-1} \hat{g}_k^2 \\ &= \sum_{k=0}^{\infty} g_k^2 - \sum_{k=0}^{K-1} g_k \hat{g}_k - \sum_{k=0}^{K-1} \hat{g}_k g_k + \sum_{k=0}^{K-1} \hat{g}_k^2 \\ &= \sum_{k=K}^{\infty} g_k^2 + \sum_{k=0}^{K-1} (g_k^2 - 2g_k \hat{g}_k + \hat{g}_k^2) \\ &= \sum_{k=0}^{K-1} (\hat{g}_k - g_k)^2 + \sum_{k=K}^{\infty} g_k^2 \end{aligned} \quad (2.44)$$

Cette erreur se décompose en deux termes. Le premier, $\sum_{k=0}^{K-1} (\hat{g}_k - g_k)^2$, dépend des estimées $\hat{g}_k$; il est d'autant plus petit que l'estimation est plus précise. Le second, $\sum_{k=K}^{\infty} g_k^2$, quant à lui, ne dépend que du choix des K filtres de la base. Evidemment, l'influence de ce terme s'affaiblit avec l'augmentation de l'ordre de troncature K . Par ailleurs, le théorème suivant démontré dans [91] donne une borne supérieure de cette erreur de modélisation.

Théorème 2.2 Soit le système H de fonction de transfert développée sous forme rationnelle suivante :

$$H(z) = \sum_{i=0}^{r-1} \frac{\alpha_i}{z - \gamma_i}, \quad |\gamma_i| < 1$$

Soit H_K son approximation réalisée à l'aide de K fonctions de la BFOR construite avec l'ensemble de pôles $\{\xi_k\}_{k=0}^{K-1}$, alors :

$$|H(e^{j\omega}) - H_K(e^{j\omega})| < \sum_{i=0}^{r-1} \left| \frac{\alpha_i}{e^{j\omega} - \gamma_i} \right| \prod_{k=0}^{K-1} \left| \frac{\gamma_i - \xi_k}{1 - \xi_k^* \gamma_i} \right| \quad (2.45)$$

On remarque que si les pôles de la base sont égaux aux pôles du système cette borne supérieure est nulle. Ce résultat implique qu'à ordre de troncature égal, si les pôles de la BFOR sont choisis proches des pôles du système, l'erreur de modélisation sera plus faible que dans le cas d'un développement sur une base RIF, de Laguerre ou de Kautz ; ces bases étant caractérisées par un seul pôle réel ou une paire de pôles complexes conjugués. En conséquence, il est nécessaire d'augmenter l'ordre de troncature du développement sur ces trois bases pour réduire l'erreur de modélisation.

Exemple 2.1 Considérons le système du second ordre suivant :

$$H(z) = \frac{1}{z + 0.8} + \frac{1}{z + 0.4}$$

La figure 2.5 montre l'évolution de cette borne supérieure en fonction de la fréquence pour des bases orthogonales à deux fonctions. Le pôle de Laguerre est ici arbitrairement choisi à -0.8 . Quant à la BFOR, nous choisissons deux bases différentes. La première a ses pôles en $\{-0.6, -0.2\}$ et la seconde en $\{-0.7, -0.3\}$. On remarque que plus les pôles des bases sont proches de ceux du système plus la borne d'erreur est petite. On remarquera aussi que le développement sur la base de Laguerre peut donner une erreur de modélisation inférieure à celle d'une BFOR lorsque les pôles de cette dernière sont plus éloignés des pôles du système.

Nous présentons dans ce qui suit différentes approches de sélection des pôles en vue de minimiser l'erreur de modélisation.

2.4.2 Choix des pôles

A chaque ordre de troncature K correspond un ou plusieurs choix optimaux des pôles de la BFOR. [97] et [82] ont établi des équations algébriques dont les solutions représentent des choix optimaux. Ces équations sont cependant difficilement utilisables en pratique. Par conséquent, nous présenterons plutôt des techniques de choix de pôles plus pratiques. Nous illustrerons, pour commencer, le cas particulier de la base de Laguerre avant de traiter le cas général.

La base de Laguerre est une base complète dans l'espace de Hardy ³ caractérisée par un pôle unique ξ . Si h est la réponse impulsionnelle du système H , son développement

³Ses séquences temporelles forment une base complète dans l'espace de Lebesgue.

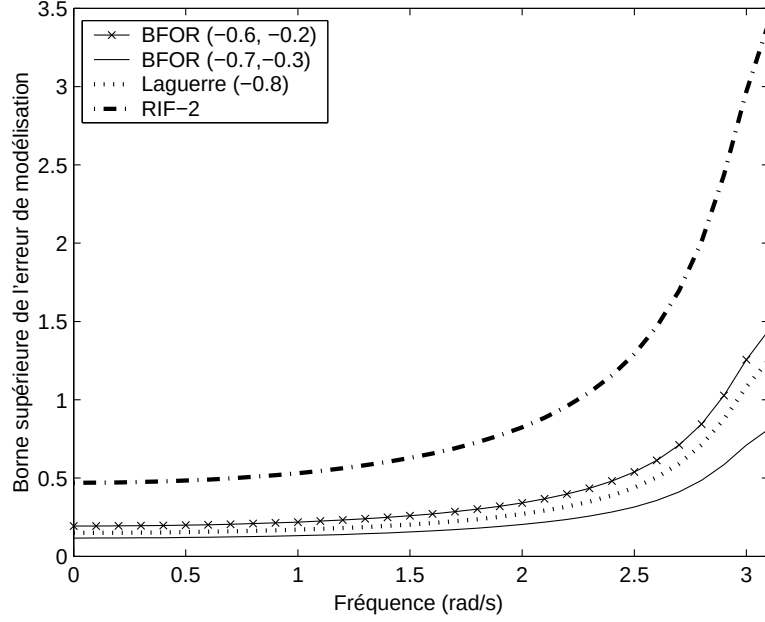


FIG. 2.5 – Borne supérieure de l'erreur de modélisation pour différents choix de pôles

sur la base de Laguerre, dans le domaine temporel, est :

$$h(n) = \sum_{k=0}^{\infty} g_k b_k(n), \quad b_k(n) = \mathcal{Z}^{-1}\{B_k(z)\}$$

2.4.2.1 Algorithme de Fu et Dumont

Dans cette procédure, le pôle optimal de Laguerre est obtenu en minimisant la fonction de coût

$$J = \sum_{k=0}^{\infty} k g_k^2 \quad (2.46)$$

Cette fonction de coût présente l'avantage d'être convexe par rapport au pôle ξ . Fu et Dumont [44] montrent qu'une solution analytique peut être trouvée à ce problème d'optimisation. En posant :

$$M_1 = \frac{1}{\|h\|^2} \sum_{n=0}^{\infty} n h^2(n), \quad M_2 = \frac{1}{\|h\|^2} \sum_{n=0}^{\infty} n (h(n+1) - h(n))^2,$$

$$\|h\|^2 = \sum_{n=0}^{\infty} h^2(n) = \sum_{k=0}^{\infty} g_k^2$$

les auteurs montrent que :

$$J = \frac{\xi M_2 + (1 - \xi)^2 M_1 + \xi - \xi^2}{1 - \xi^2} \|h\|^2$$

En écrivant que le gradient de J par rapport à ξ est égal à zéro, on obtient :

$$(M_1 + J'_{min} - 1) \xi_{opt}^2 + (M_2 - 2M_1 + 1) \xi_{opt} + M_1 - J'_{min} = 0, \quad \text{o } J'_{min} = \frac{J_{min}}{\|h\|^2} \quad (2.47)$$

Cette équation du second degré en ξ_{opt} admet une solution unique réelle lorsque son discriminant est nul. Ce qui correspond à :

$$(M_2 - 2M_1 + 1)^2 - 4(M_1 + J'_{min} - 1)(M_1 - J'_{min}) = 0$$

soit :

$$J'_{min} = \frac{1 + \sqrt{4M_1M_2 - M_2^2 - 2M_2}}{2}$$

La solution de l'équation (2.47) est alors :

$$\xi_{opt} = \frac{2M_1 - 1 - M_2}{2(M_1 + J'_{min} - 1)}$$

soit :

$$\xi_{opt} = \frac{2M_1 - 1 - M_2}{2M_1 - 1 + \sqrt{4M_1M_2 - M_2^2 - 2M_2}} \quad (2.48)$$

Ce pôle existe toujours, car on peut montrer que le terme sous la racine carrée est toujours positif et le dénominateur non nul.

Le calcul analytique de ce pôle optimal au sens de la fonction de coût (2.46) nécessite cependant la connaissance *a priori* de la réponse impulsionnelle du système. Dans le contexte de l'identification des systèmes, le calcul de la base optimale devrait donc succéder à une étape préalable d'estimation de la réponse impulsionnelle du système.

2.4.2.2 Méthode de Tanguy *et al.*

De manière similaire à celle de [44], Tanguy *et al* [126] ont trouvé une expression analytique du pôle optimal pour des systèmes non strictement causaux. Dans un tel cas, les fonctions de Laguerre sont définies par $B_k(z) = \frac{\sqrt{1-\xi^2}z}{z-\xi} \left(\frac{1-\xi z}{z-\xi}\right)^k$. La fonction de coût minimisée est :

$$J_K = \frac{\sum_{k=0}^{\infty} kg_k^2}{\|h\|^2 K} \quad (2.49)$$

On montre que

$$J_K = \frac{(1 + M_1) \xi^2 - 2m_2 \xi + M_1}{(1 - \xi^2) K}, \quad m_2 = \frac{1}{\|h\|^2} \sum_{n=1}^{\infty} nh(n)h(n-1)$$

Par suite :

$$\frac{\partial J_K}{\partial \xi} = 0 \Leftrightarrow \xi^2 - 2\rho \xi + 1 = 0, \quad \rho = \frac{2M_1 + 1}{2m_2}$$

D'où :

$$\xi_{opt} = \begin{cases} \rho - \sqrt{\rho^2 - 1}, & \text{si } \rho \geq 1 \\ \rho + \sqrt{\rho^2 - 1}, & \text{si } \rho \leq -1 \end{cases} \quad (2.50)$$

Il est à noter que cette solution ne dépend que de l'information apportée par le système via les quantités M_1 et m_2 .

Cette méthode se démarque de celle de Fu et Dumont [44] en ce qu'elle concerne une classe plus large de systèmes. En effet, la méthode de Fu et Dumont ne s'applique qu'aux systèmes ayant au moins un retard unitaire alors que celle de Tanguy et al. s'applique aux systèmes causaux ayant un retard unitaire ou non. Par ailleurs un des apports importants de Tanguy *et al.* est d'avoir exprimé le pôle optimal de Laguerre en fonction des coefficients de Laguerre. En effet, en posant :

$$\begin{aligned} S_1(\xi) &= \sum_{k=0}^{\infty} (2k+1) g_k^2(\xi) \\ S_2(\xi) &= 2 \sum_{k=1}^{\infty} k g_k(\xi) g_{k-1}(\xi) \end{aligned}$$

On montre que

$$\rho = \frac{S_2(\xi) + \alpha S_1(\xi)}{S_1(\xi) + \alpha S_2(\xi)}, \quad \alpha = \frac{1 + \xi^2}{2\xi} \quad (2.51)$$

et

$$\frac{\partial J_K}{\partial \xi} = \frac{-S_2(\xi)}{(1 - \xi^2)K \|h\|^2}$$

Cette dernière relation implique que lorsque le minimum de la fonction de coût J_K est atteint, S_2 est nul. Le pôle optimal étant unique, la réciproque est vraie, i.e. lorsque $S_2(\xi)$ s'annule ou est proche de zéro alors le pôle correspondant est le pôle optimal recherché. On peut donc à partir d'un pôle arbitrairement choisi, déterminer le spectre de Laguerre⁴ du système à identifier puis déterminer le pôle de Laguerre à l'aide de (2.51) et de (2.50).

Remarque 2.1 Les fonctions de coût minimisées par [44] et [126] sont une borne supérieure de l'erreur quadratique d'approximation (2.44). En conséquence ces deux méthodes sont sous-optimales vis-à-vis du critère de l'erreur quadratique. Il est à noter qu'une extension à une optimisation en ligne de l'algorithme de Tanguy et al. est donnée par [125].

2.4.2.3 Estimation des pôles d'une BFOR

Considérons la minimisation de la fonction de coût quadratique suivante :

$$J_K = e_K^T e_K \quad (2.52)$$

⁴Le spectre de Laguerre est constitué par l'ensemble des coefficients de Fourier du développement d'un système sur une base de Laguerre

avec $e_K = (e_K(0) \ e_K(1) \ \dots \ e_K(N-1))^T$, $e_K(n) = d(n) - \sum_{k=0}^{K-1} g_k s_k(n)$.

La minimisation de ce critère peut s'opérer en utilisant des techniques classiques d'optimisation telles que les algorithmes de type descente du gradient. Notons que la fonction de coût (2.52) est certes convexe par rapport aux coefficients de développement g_k , mais elle ne l'est pas par rapport aux pôles. En conséquence l'optimisation des pôles par les algorithmes de type descente du gradient peut présenter une convergence vers des minima locaux. En outre, si le pas d'adaptation de l'algorithme n'est pas adéquatement choisi, la convergence risque d'être lente, oscillante ou même non garantie.

Dans une fenêtre d'observation d'horizon N , l'erreur d'estimation se note :

$$e_K = d - Y = d - \Phi G \quad (2.53)$$

où :

$$\begin{aligned} d &= (d(0) \ d(1) \ \dots \ d(N-1))^T \\ Y &= (y(0) \ y(1) \ \dots \ y(N-1))^T \\ \varphi_k &= (s_k(0) \ s_k(1) \ \dots \ s_k(N-1))^T \\ \Phi &= (\varphi_0 \ \varphi_1 \ \dots \ \varphi_{K-1}) \\ G &= (g_0 \ g_1 \ \dots \ g_{K-1})^T \end{aligned}$$

φ_k et G sont des vecteurs colonne de dimensions respectives N et K et Φ est une matrice de dimension $N \times K$.

Algorithme du gradient : Cet algorithme est basé sur le développement à l'ordre 1 du critère quadratique pour des variations de pôles suffisamment petites.

$$J_K(\hat{\underline{\xi}}^{(i+1)}) = J_K(\hat{\underline{\xi}}^{(i)} + \Delta \hat{\underline{\xi}}^{(i)}) = J_K(\hat{\underline{\xi}}^{(i)}) + \left[\frac{\partial J_K(\underline{\xi})}{\partial \underline{\xi}} \right]_{\underline{\xi}=\hat{\underline{\xi}}^{(i)}}^T \Delta \hat{\underline{\xi}}^{(i)} + o(\Delta \hat{\underline{\xi}}^{(i)})$$

où $\hat{\underline{\xi}}^{(i)}$ est le vecteur de pôles à l'étape i . Pour minimiser J_K , il faut choisir $\Delta \hat{\underline{\xi}}^{(i)}$ colinéaire avec le gradient de J_K , mais dans la direction opposée. On peut donc écrire :

$$\hat{\underline{\xi}}^{(i+1)} = \hat{\underline{\xi}}^{(i)} - \mu \left[\frac{\partial J_K(\underline{\xi})}{\partial \underline{\xi}} \right]_{\underline{\xi}=\hat{\underline{\xi}}^{(i)}} \quad (2.54)$$

On a :

$$\begin{aligned} \frac{\partial J_K(\underline{\xi})}{\partial \underline{\xi}} &= 2 \frac{\partial e_K^T}{\partial \underline{\xi}} e_K \\ \frac{\partial e_K^T}{\partial \underline{\xi}} &= - \left(\frac{\partial \Phi}{\partial \xi_0} G \ \frac{\partial \Phi}{\partial \xi_1} G \ \dots \ \frac{\partial \Phi}{\partial \xi_{K-1}} G \right)^T \\ \frac{\partial \Phi}{\partial \xi_k} &= \left(\frac{\partial \varphi_0}{\partial \xi_k} \ \frac{\partial \varphi_1}{\partial \xi_k} \ \dots \ \frac{\partial \varphi_{K-1}}{\partial \xi_k} \right) \end{aligned} \quad (2.55)$$

2.4. Choix de la base optimale

On peut remarquer que si on suppose que $\xi_i \neq \xi_k$ pour $i > k$ alors : $\frac{\partial \varphi_i}{\partial \xi_k} = 0$, $k = i + 1, \dots, K - 1$. Par suite, l'équation (2.55) devient :

$$\frac{\partial e_K^T}{\partial \underline{\xi}} = \left(-\frac{\partial \Phi}{\partial \xi_0} G - \left(0_{N \times 1} \frac{\partial \varphi_1}{\partial \xi_1} \quad \dots \quad \frac{\partial \varphi_{K-1}}{\partial \xi_1} \right) G \quad \dots \quad - \left(0_{N \times 1} \quad \dots \quad 0_{N \times 1} \frac{\partial \varphi_{K-1}}{\partial \xi_{K-1}} \right) G \right)^T$$

Une expression complète de l'algorithme est obtenue en calculant les différentes sensibilités $\frac{\partial \varphi_i}{\partial \xi_j}$. Pour ce faire, on doit calculer les dérivées des fonctions de base par rapport aux pôles. Les fonctions de base considérées ici étant :

$$B_0(z) = \frac{\sqrt{1 - \xi_0^2}}{z - \xi_0}, \quad B_k(z) = \frac{\sqrt{1 - \xi_k^2}}{z - \xi_k} \prod_{i=0}^{k-1} \frac{1 - \xi_i z}{z - \xi_i}, \quad k > 0$$

on a alors :

$$\begin{aligned} \frac{\partial B_k(z, \underline{\xi})}{\partial \xi_i} &= 0, \quad i > k \\ \frac{\partial B_k(z, \underline{\xi})}{\partial \xi_k} &= \frac{1 - \xi_k z}{(1 - \xi_k^2)(z - \xi_k)} B_k(z, \underline{\xi}) \\ \frac{\partial B_k(z, \underline{\xi})}{\partial \xi_i} &= \frac{1 - z^2}{(1 - \xi_i z)(z - \xi_i)} B_k(z, \underline{\xi}), \quad i < k \end{aligned}$$

Cette dernière équation contient un pôle instable en $z = 1/\xi_i$ qui est théoriquement annulé par un zéro de la fonction de base. Toutefois, en pratique, toute erreur numérique provenant du calcul en virgule flottante peut causer la non-annulation de ce pôle. D'où la nécessité de prendre le soin d'éliminer le pôle et le zéro concernés avant toute implémentation de l'équation.

Algorithme de Gauss-Newton : En vue d'améliorer la rapidité de convergence de l'algorithme d'estimation des pôles, on peut utiliser les algorithmes de type Newton qui ont des pas adaptatifs. Ils sont basés sur le développement à l'ordre 2 du critère quadratique. On a alors :

$$\underline{\xi}^{(i+1)} = \underline{\xi}^{(i)} - \mu \left[\frac{\partial^2 J_K(\underline{\xi})}{\partial \underline{\xi}^T \partial \underline{\xi}} \right]_{\underline{\xi} = \underline{\xi}^{(i)}}^{-1} \left[\frac{\partial J_K(\underline{\xi})}{\partial \underline{\xi}} \right]_{\underline{\xi} = \underline{\xi}^{(i)}} \quad (2.56)$$

Le paramètre μ a pour rôle d'atténuer ou d'amplifier l'effet de l'inverse du Hessien $\left[\frac{\partial^2 J_K(\underline{\xi})}{\partial \underline{\xi}^T \partial \underline{\xi}} \right]^{-1}$. Généralement, le choix $\mu = 1$ correspond au pas optimal qui conduit au minimum de l'erreur quadratique en un nombre réduit d'itérations [82]. Une forte contrainte à l'utilisation de cette méthode est la nécessité d'avoir la matrice Hessienne définie non négative. C'est le cas lorsqu'on s'approche de l'optimum mais généralement pas au début de la procédure. Par ailleurs, une expression du Hessien est :

$$\frac{\partial^2 J_K}{\partial \underline{\xi}^T \partial \underline{\xi}} = \frac{\partial e_K}{\partial \underline{\xi}} \frac{\partial e_K^T}{\partial \underline{\xi}} + 2 \frac{\partial^2 e_K}{\partial \underline{\xi}^T \partial \underline{\xi}} e_K \quad (2.57)$$

A l'approche de l'optimum, les signaux d'erreur contenus dans le vecteur e_K sont très petits. Le second terme du membre de droite de (2.57) est donc négligeable. D'où l'approximation :

$$\frac{\partial^2 J_K^{(app)}}{\partial \underline{\xi}^T \partial \underline{\xi}} = \frac{\partial e_K}{\partial \underline{\xi}} \frac{\partial e_K^T}{\partial \underline{\xi}} \quad (2.58)$$

La relation (2.58) donne le Hessien approché qui est défini positif. Cette propriété assure ainsi la possibilité de toujours trouver un nouveau vecteur de pôles qui diminue le critère. L'évaluation du Hessien approché ne nécessite que la connaissance des sensibilités du premier ordre déjà calculées pour le gradient.

Remarque 2.2 Dans le cas particulier de la base de Laguerre, la méthode sous-optimale de Tanguy et al. peut être utilisée pour initialiser un algorithme de type descente du gradient. Une telle initialisation permet d'éviter certains minima locaux [85].

2.5 Choix de la meilleure base

2.5.1 Effet de l'ordre de placement des filtres passe-tout sur le choix de la base

Une fois les pôles choisis, il y a différentes manières d'ordonner les filtres passe-tout dont la mise en cascade génère la BFOR. [16] ont montré que suivant un critère de compression de données, on peut, par la suite, choisir la base la plus efficace possible.

Considérons l'ensemble de filtres passe-tout suivant $\left\{ G_i(z) = \frac{1 - \xi_i^* z}{z - \xi_i} \right\}_{i=0}^{K-1}$. Rappelons qu'à chaque filtre passe-tout G_i est associé le filtre passe-bas $V_i(z) = \frac{\sqrt{1 - |\xi_i|^2}}{z - \xi_i}$ et que les fonctions de base générées par la mise en cascade de ces passe-tout sont $B_k(z) = V_k(z) \prod_{i=0}^{k-1} G_i(z)$. On peut remarquer qu'une telle fonction est caractérisée par le dernier filtre de la cascade, à savoir G_k ou mieux par le passe-bas V_k qui lui est associé, et par l'ensemble des filtres en amont, i.e. les G_i , $i = 0, 1, \dots, k-1$. Chaque passe-tout étant caractérisé par un pôle unique, se donner un ordre de placement des pôles revient à ordonner les filtres passe-tout dans la cascade qui génère les fonctions de base.

Considérons un certain ordre de placement des K pôles ξ_i . Nous pouvons alors générer un ensemble de fonctions constituant une base $\mathfrak{B}_0 = \{B_{0;0}(z), \dots, B_{K-1;0}(z)\}$. En réalisant toutes les permutations possibles des pôles considérés, il est possible de former $K!$ bases $\mathfrak{B}_i$ différentes⁵. Si les fonctions de base $B_{k;i}$ étaient toutes différentes, on aurait $K!K$ différentes fonctions. Cependant, la proposition suivante montre que l'intersection entre les bases $\mathfrak{B}_i$ n'est pas vide ; il y a des fonctions qui sont communes à différentes bases.

⁵Dans ce paragraphe, on notera $B_{k;i}$ la $(k+1)$ -ème fonction de la base $\mathfrak{B}_i$

Proposition 2.3 *Etant donné K filtres passe-tout distincts, il y a $K2^{K-1}$ fonctions distinctes formant les $K!$ bases possibles.*

Preuve : On désire générer des bases avec l'ensemble de filtres passe-tout $\{G_i\}_{i=0}^{K-1}$. Il y a $K!$ bases possibles. Chaque fonction de base est caractérisée par le dernier filtre de la cascade G_k (représenté par V_k) et par l'ensemble des filtres en amont dans la cascade. Cet ensemble est adressé par le K -uplet $(q_0, \dots, q_{K-1})$ où $q_i = 1$ signifie la présence du filtre G_i dans l'ensemble des filtres précédant G_k et $q_i = 0$ son absence. A toute fonction de base caractérisée par G_k comme dernier filtre de la cascade, on associe le K -uplet $(q_0, \dots, q_{k-1}, 1, q_{k+1}, \dots, q_{K-1})$. En d'autres termes, on associe, de façon univoque, un nombre binaire pouvant prendre 2^{K-1} valeurs à toute fonction de base caractérisée par G_k comme dernier filtre de la cascade. Il y a donc 2^{K-1} fonctions de base caractérisées par G_k comme dernier filtre de la cascade. Sachant qu'il y a K possibilités différentes de choisir le filtre G_k , le nombre total de fonctions distinctes s'élève donc à $K2^{K-1}$ ■

Exemple 2.2 *Pour illustrer la proposition ci-dessus, considérons l'ensemble de filtres passe-tout $\{G_i\}_{i=0}^2$. Nous pouvons construire 6 bases différentes de trois fonctions chacune. Dans le tableau 2.1 nous donnons les différentes fonctions correspondant à chacune des bases. On peut aisément remarquer qu'il y a des fonctions qui sont communes à des bases différentes. En faisant un décompte des fonctions, le nombre total des fonctions est bien égal à 12.*

Base i	Fonction $B_{0;i}(z)$	Fonction $B_{1;i}(z)$	Fonction $B_{2;i}(z)$
$\mathfrak{B}_0$	$V_0(z)$	$V_1(z)G_0(z)$	$V_2(z)G_0(z)G_1(z)$
$\mathfrak{B}_1$	$V_1(z)$	$V_0(z)G_1(z)$	$V_2(z)G_0(z)G_1(z)$
$\mathfrak{B}_2$	$V_0(z)$	$V_2(z)G_0(z)$	$V_1(z)G_0(z)G_2(z)$
$\mathfrak{B}_3$	$V_2(z)$	$V_0(z)G_2(z)$	$V_1(z)G_0(z)G_2(z)$
$\mathfrak{B}_4$	$V_1(z)$	$V_2(z)G_1(z)$	$V_0(z)G_1(z)G_2(z)$
$\mathfrak{B}_5$	$V_2(z)$	$V_1(z)G_2(z)$	$V_0(z)G_1(z)G_2(z)$

TAB. 2.1 – Bases générées par la mise en cascade de l'ensemble de passe-tout $\{G_i\}_{i=0}^2$

Les critères de sélection de la meilleure base parmi ces $K!$, que nous présenterons plus loin, étant liés aux coefficients de Fourier, une étape d'estimation de ces coefficients est donc nécessaire. Pour ce faire, considérons que nous avons réalisé l'estimation des coefficients de Fourier $g_{k;0}$ associés aux fonctions de la base $\mathfrak{B}_0$ ⁶. Selon [13, 17], il est possible d'obtenir les coefficients de Fourier associés à toutes les $K2^{K-1}$ fonctions à partir de ceux associés aux fonctions de $\mathfrak{B}_0$.

⁶ $g_{k;i}$ est le coefficient de Fourier associé à la $(k+1)$ -ème fonction de la $(i+1)$ -ème base.

Considérons les deux fonctions consécutives suivantes appartenant à la base $\mathfrak{B}_l$:

$$B_{k;l}(z) = V_k(z) \prod_{i=0}^{k-1} G_i(z)$$

$$B_{k+1;l}(z) = V_{k+1}(z) \prod_{i=0}^k G_i(z) = V_{k+1}(z) G_k(z) \prod_{i=0}^{k-1} G_i(z)$$

Elles résultent de la représentation en cascade ci-dessous :

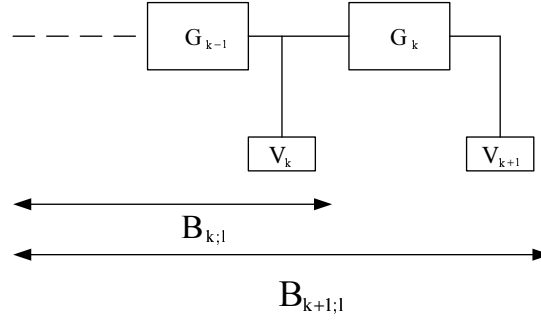


FIG. 2.6 – Cascade de filtres générant les fonctions de base $B_{k;l}$ et $B_{k+1;l}$

En intervertissant les deux derniers pôles, on obtient la cascade suivante :

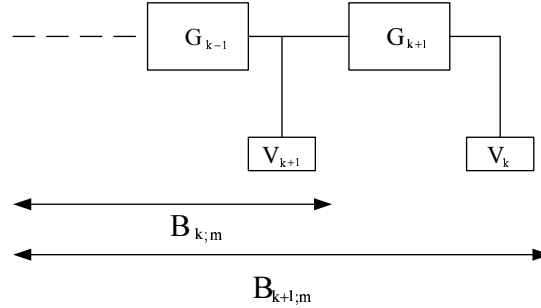


FIG. 2.7 – Cascade de filtres après interversion des pôles

Celle-ci génère deux fonctions d'une autre base $\mathfrak{B}_m$:

$$B_{k;m}(z) = V_{k+1}(z) \prod_{i=0}^{k-1} G_i(z)$$

$$B_{k+1;m}(z) = V_k(z) G_{k+1}(z) \prod_{i=0}^{k-1} G_i(z)$$

On note $\mathcal{T}_{l,m}(k, k+1)$ la matrice de passage entre les deux bases :

$$\begin{pmatrix} B_{k;l} \\ B_{k+1;l} \end{pmatrix} = \mathcal{T}_{l,m}(k, k+1) \begin{pmatrix} B_{k;m} \\ B_{k+1;m} \end{pmatrix}$$

avec

$$\mathcal{T}_{l,m}(k, k+1) = \begin{pmatrix} \langle B_{k;l}, B_{k;m} \rangle & \langle B_{k;l}, B_{k+1;m} \rangle \\ \langle B_{k+1;l}, B_{k;m} \rangle & \langle B_{k+1;l}, B_{k+1;m} \rangle \end{pmatrix}$$

En éliminant le facteur commun aux quatre fonctions de base, à savoir $\prod_{i=0}^{k-1} G_i(z)$, l'expression de la matrice de passage $\mathcal{T}_{l,m}(k, k+1)$ devient :

$$\mathcal{T}_{l,m}(k, k+1) = \begin{pmatrix} \langle V_k(z), V_{k+1}(z) \rangle & \langle V_k(z), V_k(z)G_{k+1}(z) \rangle \\ \langle V_{k+1}(z)G_k(z), V_{k+1}(z) \rangle & \langle V_{k+1}(z)G_k(z), V_k(z)G_{k+1}(z) \rangle \end{pmatrix}$$

D'une manière générale on a [14] :

$$\mathcal{T}_{l,m}(k, j) = \begin{pmatrix} \langle V_k(z), V_j(z) \rangle & \langle V_k(z), V_k(z)G_j(z) \rangle \\ \langle V_j(z)G_k(z), V_j(z) \rangle & \langle V_j(z)G_k(z), V_k(z)G_j(z) \rangle \end{pmatrix} \quad (2.59)$$

Le calcul de ces différents produits scalaires dans le cas des pôles réels donne :

$$\mathcal{T}_{l,m}(k, j) = \frac{1}{1 - \xi_j \xi_k} \begin{pmatrix} \sqrt{(1 - \xi_k^2)(1 - \xi_j^2)} & \xi_k - \xi_j \\ \xi_j - \xi_k & \sqrt{(1 - \xi_k^2)(1 - \xi_j^2)} \end{pmatrix} \quad (2.60)$$

Remarque 2.3 Dans le cas des pôles réels, la transformation $\mathcal{T}_{l,m}(k, j)$ est orthogonale. On vérifie aisément que

$$\mathcal{T}_{l,m}(k, j) \mathcal{T}_{l,m}^T(k, j) = I$$

Considérons les développements respectifs du système H sur les deux bases construites ci-dessus. On a : $g_{k;l} = \langle H, B_{k;l} \rangle$. On peut donc aisément vérifier que grâce à la transformation $\mathcal{T}_{l,m}(k, k+1)$, on a :

$$\begin{pmatrix} g_{k;m} \\ g_{k+1;m} \end{pmatrix} = \mathcal{T}_{l,m}^T(k, k+1) \begin{pmatrix} g_{k;l} \\ g_{k+1;l} \end{pmatrix}$$

Ainsi, connaissant deux coefficients de Fourier associés aux fonctions d'une base donnée, la matrice de transformation $\mathcal{T}$ permet de déterminer deux coefficients de Fourier associés aux fonctions d'une autre base. A partir des coefficients de Fourier, estimés, associés à une base obtenue avec un arrangement de pôles donné, on peut donc déterminer les coefficients de Fourier associés aux fonctions de toutes les $K!$ bases possibles en utilisant les transformations $\mathcal{T}$ appropriées.

Pour toutes les bases, l'énergie du système approximé est la même, i.e $\sum_{k=0}^{K-1} |g_k|^2$ est une constante (pour K suffisamment élevé). Cependant, la norme ℓ_1 des coefficients de

Fourier peut être différente d'une base à l'autre. Elle constitue une mesure de la compression de l'information contenue dans les coefficients de Fourier. Une autre mesure de la compression est la fonction entropie définie telle que :

$$Ent_i = - \sum_{k=0}^{K-1} |g_k|^2 \log |g_k| \quad (2.61)$$

La meilleure base est donc choisie en minimisant ce critère de compression. En effet, l'information significative sur le système n'est alors contenue que dans les premiers coefficients de Fourier. Les coefficients les moins significatifs peuvent alors être facilement seuillés. On peut trouver un algorithme rapide de sélection de la meilleure base au sens de l'entropie minimale dans [16].

2.5.2 Technique de seuillage

Une fois la base fixée, la complexité du modèle peut être réduite en supprimant les fonctions de base ayant des coefficients de Fourier suffisamment petits. Cela suppose donc de réaliser une étape d'estimation des coefficients de Fourier avant d'appliquer le seuillage. Les techniques d'estimation paramétrique classiques peuvent à cet effet être utilisées. Grâce à l'orthogonalité des fonctions de base, une fois qu'un ou plusieurs coefficients de Fourier ont été retranchés, il n'est pas nécessaire d'estimer de nouveau les coefficients restants.

Dans une fenêtre d'observation de largeur N , pour un signal d'entrée de densité spectrale constante égale à σ_u^2 et un bruit gaussien, de moyenne nulle et de variance σ^2 , une règle de seuillage suggérée par Donoho [13] est la suivante :

$$g_k^s = \begin{cases} g_k & \text{si } |g_k| \geq \frac{\sigma}{\sigma_u} \sqrt{\frac{2 \log K}{N}} \\ 0, & \text{sinon} \end{cases}, k = 0, 1, \dots, K-1 \quad (2.62)$$

2.6 Conclusion

Différentes approches ont été présentées dans ce chapitre pour la synthèse d'une base de fonctions orthogonales rationnelles. Toutes les approches peuvent être résumées en ceci : la mise en cascade de filtres passe-tout à représentation équilibrée dans l'espace d'état permet de générer une BFOR. Dans la suite de notre travail, cependant, nous considérerons les fonctions de base issues de la construction de Ninness et Gustafsson (1997).

Le choix des pôles est une question cruciale vis-à-vis de l'ordre de troncature du développement sur une BFOR utilisé pour la modélisation des systèmes. Plus les pôles de

la base sont proches des dynamiques dominantes du système, plus la représentation du système se fait avec un nombre réduit de paramètres. Nous avons présenté différentes approches pour effectuer le choix des pôles. En général, il est nécessaire d'avoir une connaissance, fut-elle partielle, du système.

La BFOR étant complète dans l'espace de Hardy, il est toujours possible de fixer arbitrairement les pôles. Toutefois, un tel choix peut conduire à des ordres de troncature de l'approximation élevés. Or, on sait que lorsque le nombre de paramètres à estimer est élevé, la variance de l'erreur d'estimation l'est aussi. L'approche d'optimisation des pôles semble être souhaitable bien que sa convergence ne soit garantie que vers des minima locaux. Dans la littérature, nous n'avons pas trouvé d'application de cette technique sur des pôles complexes. Ceci est certainement dû à la complexité des équations induites [82]. Pour améliorer la parcimonie de la représentation, on peut associer les techniques d'optimisation, de choix de la meilleure base suivant un critère de compression de données et de seuillage. Nous proposons dans le chapitre suivant un algorithme qui réalise ces trois tâches de façon implicite.

Chapitre 3

Réduction de complexité à l'aide de bases de fonctions orthogonales rationnelles

3.1 Introduction

Le principe d'utilisation des bases orthogonales pour représenter les noyaux de Volterra n'est pas nouveau en soi. Déjà, en 1958, Wiener a proposé l'utilisation d'une base multidimensionnelle formée à partir des fonctions de Laguerre. Dans ce chapitre, nous appliquons ce principe pour d'autres bases généralisant la base de Laguerre et traitons le problème du choix de la dite base. Certains auteurs ont déjà utilisé les bases de type BFOR comme bases de développement des noyaux de Volterra [54, 55, 49, 35, 36]. Cependant seuls Hacıoglu et Williamson [54, 55] ont traité la question du choix de la base mais dans le cas particulier des modèles de Wiener. Leur algorithme se base sur la connaissance de la réponse impulsionnelle de la partie linéaire du modèle. C'est une extension de la méthode de Steiglitz-Mc-Bride [138].

L'objet de ce chapitre est de présenter de nouveaux outils d'optimisation de bases pour le développement des noyaux de Volterra.

Dans le chapitre précédent, nous avons montré que dans le cas linéaire, un développement sur une base de Laguerre est généralement moins parcimonieux que sur une BFOR. Cependant, l'optimisation d'une base de Laguerre est plus "facile" que celle d'une BFOR. La base de Laguerre étant caractérisée par un seul paramètre, les méthodes de Fu et Dumont [44] et de Tanguy *et al* [126, 125] permettent d'optimiser un critère convexe et donc d'obtenir une solution réelle unique. Pour l'optimisation de la BFOR, seules des méthodes sous-optimales existent. Après avoir présenté le principe

du développement des noyaux de Volterra sur une base orthogonale, nous commencerons par développer une méthode d'optimisation de la base de Laguerre avant de présenter celles concernant la BFOR.

3.2 Principe du développement des noyaux de Volterra sur une base orthogonale

Nous allons montrer comment effectuer le développement d'un noyau d'ordre p sur une base orthogonale multidimensionnelle formée à partir des fonctions d'une base orthogonale monodimensionnelle. Dans le chapitre 2, nous avons montré que, sous les hypothèses de stabilité et de causalité, le développement du noyau de Volterra du premier ordre sur une base orthogonale $\mathfrak{B} = \{b_k\}_{k=0}^{\infty}$ est :

$$h_1(i) = \sum_{k=0}^{\infty} g_k b_k(i)$$

où les g_k sont les coefficients de Fourier. Considérons à présent un noyau quadratique stable et causal de coefficients $h_2(n_1, n_2)$. Si nous fixons l'un des indices, par exemple n_2 , nous retrouvons une fonction du monodimensionnelle. Elle peut être développée sur la base $\mathfrak{B}$ de sorte que :

$$h_2(n_1, n_2) = \sum_{k_1=0}^{\infty} \alpha_{k_1}(n_2) b_{k_1}(n_1)$$

avec $\alpha_{k_1}(n_2) = \sum_{n_1=0}^{\infty} h_2(n_1, n_2) b_{k_1}(n_1)$.

Les coefficients $\alpha_{k_1}(n_2)$ sont fonction de n_2 puisque pour différentes valeurs de n_2 , $h_2(n_1, n_2)$ est généralement une fonction différente de n_1 . Nous pouvons alors développer α_{k_1} sur la base $\mathfrak{B}$. On a :

$$\alpha_{k_1}(n_2) = \sum_{k_2=0}^{\infty} g_{k_1, k_2} b_{k_2}(n_2)$$

avec

$$g_{k_1, k_2} = \sum_{n_2=0}^{\infty} \alpha_{k_1}(n_2) b_{k_2}(n_2)$$

Par suite :

$$h_2(n_1, n_2) = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} g_{k_1, k_2} b_{k_1}(n_1) b_{k_2}(n_2)$$

et

$$g_{k_1, k_2} = \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} h_2(n_1, n_2) b_{k_1}(n_1) b_{k_2}(n_2)$$

3.2. Principe du développement des noyaux de Volterra sur une base orthogonale

En étendant ce raisonnement aux noyaux d'ordre supérieur on montre que d'une manière générale, le développement d'un noyau d'ordre p sur une base orthogonale $\mathfrak{B}_p = \{b_{k,p}\}_{k=0}^\infty$ est [121] :

$$h_p(n_1, \dots, n_p) = \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} g_{k_1, \dots, k_p} b_{k_1,p}(n_1) \cdots b_{k_p,p}(n_p) \quad (3.1)$$

avec

$$g_{k_1, \dots, k_p} = \sum_{n_1=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} h_p(n_1, \dots, n_p) b_{k_1,p}(n_1) \cdots b_{k_p,p}(n_p) \quad (3.2)$$

Le développement des noyaux de Volterra sur une base orthogonale conduit à une représentation à l'aide d'un banc de filtres orthogonaux telle qu'illustrée par la figure 3.1.

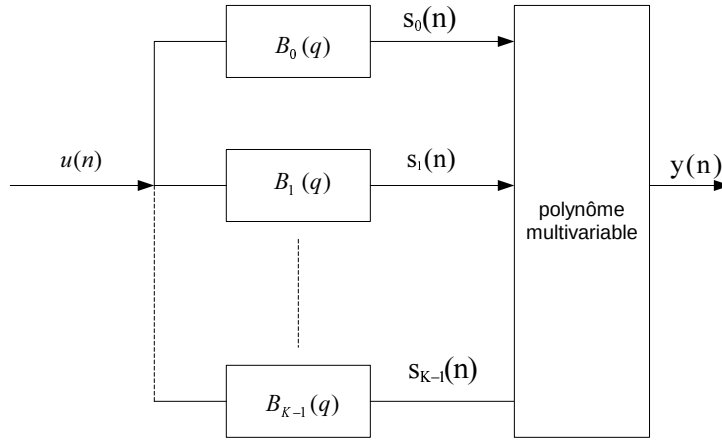


FIG. 3.1 – Modèle de Volterra représenté à l'aide d'un banc de filtres orthogonaux

En reprenant le raisonnement ci-dessus dans le cas d'un noyau séparable, il est aisé de montrer que :

$$h_p(n_1, \dots, n_p) = \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} g_{k_1} \cdots g_{k_p} b_{k_1,p}(n_1) \cdots b_{k_p,p}(n_p)$$

En conséquence, la propriété de séparabilité du noyau est reportée sur les coefficients de Fourier. A partir de la relation (3.1) on a en effet :

$$g_{k_1, \dots, k_p} = \prod_{j=1}^p g_{k_j} \quad (3.3)$$

Il est à noter qu'en pratique nous utiliserons des développements à un ordre fini K . Dans ce cas, le développement (3.1) requiert K^p paramètres. En rappelant que si le modèle est de mémoire M le noyau h_p est constitué de M^p paramètres, tronquer le développement à un ordre K tel que $K \ll M$ permet de réduire considérablement la complexité

du modèle. Par ailleurs, il est intéressant de constater que si le noyau est supposé symétrique, les coefficients de Fourier le sont aussi. On peut donc leur appliquer la technique de triangularisation des noyaux. En conséquence le nombre de coefficients de Fourier est réduit à $\frac{(K+p-1)!}{(K-1)!p!}$.

Remarque 3.1 *De la même manière que pour les coefficients des noyaux de Volterra, nous pouvons organiser les coefficients de Fourier suivant une écriture vectorielle, matricielle ou tensorielle. En conséquence, on peut aussi leur appliquer les techniques de décomposition matricielle et tensorielle que nous avons présentées (voir le paragraphe 1.4).*

Remarque 3.2 *Dans le cas d'un noyau séparable, il est intéressant de remarquer, à partir de (3.3), que le développement du noyau sur une base à K fonctions n'est en fait caractérisé que par K coefficients g_k qui permettent de générer l'ensemble des coefficients de Fourier $g_{k_1, \dots, k_p}$.*

Après développement sur une base orthogonale, sous l'hypothèse de symétrie du noyau initial, le modèle résultant s'écrit :

$$\begin{aligned} y(n) &= \sum_{p=1}^P y_p(n) \\ y_p(n) &= \sum_{k_1=0}^{K-1} \cdots \sum_{k_p=k_{p-1}}^{K-1} g_{k_1, \dots, k_p} \prod_{j=1}^p s_{k_j, p}(n) \end{aligned} \quad (3.4)$$

$$\text{où } s_{k_j, p}(n) = \sum_{i=0}^{M-1} b_{k_j, p}(i) u(n-i).$$

3.3 Modèle de Volterra-Laguerre

Campello *et al* ont été les premiers à s'intéresser à la question du choix du pôle optimal de Laguerre dans le cas d'un modèle de Volterra. Ils ont étendu l'algorithme de Fu et Dumont [44] au cas d'un modèle de Volterra [20, 21]. Dans leur travail, ils étudient le développement des noyaux aussi bien sur des bases différentes que sur une base unique. Ils montrent à cet effet l'avantage d'un développement sur des bases différentes plutôt que sur une base unique. Cependant, l'approche nécessite la connaissance des noyaux d'où la nécessité de leur estimation avant toute optimisation des pôles. Campello [19] propose la technique itérative suivante :

1. Choix d'un pôle initial ;
2. Estimation des coefficients de Fourier ;
3. Reconstruction des noyaux à partir des coefficients de Fourier ;

4. Calcul du pôle optimal et retour à l'étape 2 ;

En général l'algorithme converge en quelques itérations. Une telle technique se réalise dans une estimation *off-line*. Nous avons proposé un algorithme adaptatif alliant l'algorithme *RLS* et le calcul analytique du pôle optimal dans le cas d'un modèle quadratique [67, 69]. Cet algorithme recourt aussi à la reconstruction du noyau à chaque itération. Ce qui augmente, du reste, la complexité de calcul de l'algorithme.

Il est à noter par ailleurs que l'algorithme de Campello ne s'applique que si le noyau a au moins un retard unitaire (ou encore s'il est strictement causal) , i.e. $h_{n_1, \dots, n_p} = 0$ si $\exists i \in 1, \dots, p : n_i = 0$. Or il s'avère que bien des modèles de Volterra discrets, rencontrés dans la littérature, ne remplissent pas ce critère. Nous présentons donc dans ce qui suit, une approche plus générale qui peut-être vue comme une extension de l'algorithme de Tangue et al [125].

Les fonctions de Laguerre utilisées sont telles que :

$$B_{k,p}(z) = \frac{\sqrt{1 - \xi_p^2} z}{z - \xi_p} \left(\frac{1 - \xi_p z}{z - \xi_p} \right)^k \quad (3.5)$$

Elles vérifient les deux relations suivantes [126, 125] :

$$kb_{k,p}(i) = -\frac{(i+1)\xi_p}{1 - \xi_p^2} b_{k,p}(i+1) + \frac{i(1 + \xi_p^2) + \xi_p^2}{1 - \xi_p^2} b_{k,p}(i) - \frac{i\xi_p}{1 - \xi_p^2} b_{k,p}(i-1) \quad (3.6)$$

$$\frac{\partial b_{k,p}(i)}{\partial \xi_p} = \frac{1}{1 - \xi_p^2} ((k+1) b_{k+1,p}(i) - kb_{k-1,p}(i)) \quad (3.7)$$

Nous allons considérer l'optimisation séparée des bases relatives à chaque noyau. Pour ce faire, considérons la fonction de coût suivante :

$$J_p = \frac{1}{p \|h_p\|^2} \sum_{k_1=0}^{\infty} \dots \sum_{k_p=0}^{\infty} (k_1 + \dots + k_p) g_{k_1, \dots, k_p}^2 \quad (3.8)$$

avec $\|h_p\|^2 = \sum_{n_1=0}^{\infty} \dots \sum_{n_p=0}^{\infty} h_p^2(n_1, \dots, n_p)$. On peut remarquer qu'il s'agit de la même fonction de coût que celle proposée par [21]. Elle est une borne supérieure de l'erreur quadratique de modélisation due à la troncature à un ordre fini du développement de Laguerre, résultat prouvé dans [21].

Définissons les quantités suivantes :

$$M_{1,l} = \frac{1}{\|h_p\|^2} \sum_{n_1=0}^{\infty} \dots \sum_{n_p=0}^{\infty} n_l h_p^2(n_1, \dots, n_p), \quad l = 1, \dots, p \quad (3.9)$$

$$M_{2,l} = \frac{1}{\|h_p\|^2} \sum_{n_1=0}^{\infty} \dots \sum_{n_p=0}^{\infty} n_l h_p(n_1, \dots, n_l, n_{l+1}, \dots, n_p) \\ \times h_p(n_1, \dots, n_l - 1, n_{l+1}, \dots, n_p), \quad l = 1, \dots, p \quad (3.10)$$

$$Q_{j,p} = \frac{1}{p} \sum_{l=1}^p M_{j,l}, \quad j = 1, 2 \quad (3.11)$$

La fonction de coût (3.8) peut alors s'exprimer de manière explicite en fonction du pôle de Laguerre :

Proposition 3.1 *Sous l'hypothèse de stabilité du modèle de Volterra, l'égalité suivante est vérifiée :*

$$J_p = \frac{(1 + Q_{1,p}) \xi_p^2 - 2Q_{2,p} \xi_p + Q_{1,p}}{(1 - \xi_p^2)} \quad (3.12)$$

Preuve : voir l'annexe B.1 ■

Notons que le numérateur de J_p est une fonction convexe, différentiable et non-négative définie sur un ouvert convexe $\mathcal{P} = \{\xi_p \in \mathfrak{R} : |\xi_p| < 1\}$ tandis que son dénominateur est une fonction concave, différentiable et positive sur $\mathcal{P}$. En conséquence, J_p est une fonction pseudo-convexe sur $\mathcal{P}$. Toute solution de $\frac{\partial J_p}{\partial \xi_p} = 0$ est donc un minimum global de J_p [7]. En vue de déterminer ce minimum, posons :

$$\rho_{0,p} = \frac{2Q_{1,p} + 1}{2Q_{2,p}} \quad (3.13)$$

Par suite :

Proposition 3.2 *Le pôle optimal de Laguerre, au sens du critère (3.8), est :*

$$\xi_{p,opt} = \begin{cases} \rho_{0,p} - \sqrt{\rho_{0,p}^2 - 1}, & \text{si } \rho_{0,p} > 1 \\ \rho_{0,p} + \sqrt{\rho_{0,p}^2 - 1}, & \text{si } \rho_{0,p} < -1 \end{cases} \quad (3.14)$$

Preuve : voir l'annexe B.2 ■

Une observation intéressante est que le pôle optimal, ainsi déterminé, ne dépend pas de l'ordre de troncature du développement du noyau sur la base de Laguerre.

3.3.1 Expression du pôle optimal en fonction du spectre de Laguerre

Le pôle optimal tel que nous l'avons déterminé est fonction de $\rho_{0,p}$ qui dépend du noyau de Volterra ($\rho_{0,p}$ est une caractéristique du système). En conséquence, il faut donc passer par une estimation du noyau avant de déterminer le pôle optimal. Dans ce

paragraphe, nous nous proposons d'exprimer la quantité $\rho_{o,p}$ directement en fonction des coefficients de Fourier du développement du noyau de Volterra sur une base de Laguerre quelconque. Une telle expression nous permettra de déterminer le pôle optimal directement à partir des coefficients de Fourier sans calculer les coefficients du noyau de Volterra.

Pour ce faire, définissons les quantités suivantes :

$$T_{1,l} = \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) g_{k_1, \dots, k_p}^2, \quad l = 1, \dots, p \quad (3.15)$$

$$T_{2,l} = 2 \sum_{k_1=0}^{\infty} \cdots \sum_{k_{l-1}=0}^{\infty} \sum_{k_l=1}^{\infty} \sum_{k_{l+1}=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_{l-1}, k_l, k_{l+1}, \dots, k_p} \times g_{k_1, \dots, k_{l-1}, k_l-1, k_{l+1}, \dots, k_p}, \quad l = 1, \dots, p \quad (3.16)$$

$$R_{j,p} = \sum_{l=1}^p T_{j,l}, \quad j = 1, 2 \quad (3.17)$$

Avant d'exprimer $Q_{j,p}$, $j = 1, 2$ en fonction de $R_{j,p}$, nous pouvons remarquer que la propriété (3.7) combinée à la formule (3.2) permet d'écrire ce qui suit :

$$\frac{\partial}{\partial \xi_p} g_{k_1, \dots, k_p} = \frac{1}{1 - \xi_p^2} \sum_{l=1}^p ((k_l + 1) g_{k_1, \dots, k_{l-1}, k_l+1, \dots, k_p} - k_l g_{k_1, \dots, k_{l-1}, k_l-1, \dots, k_p})$$

Grâce à cette relation nous avons le lemme suivant :

Lemme 3.1 Les quantités $R_{1,p}$ et $R_{2,p}$ sont reliées via leur dérivées par rapport au pôle ξ_p par :

$$\frac{\partial R_{1,p}}{\partial \xi_p} = \frac{-2}{1 - \xi_p^2} R_{2,p} \quad (3.18)$$

$$\frac{\partial R_{2,p}}{\partial \xi_p} = \frac{-2}{1 - \xi_p^2} R_{1,p} \quad (3.19)$$

Preuve : voir l'annexe B.3 ■

A partir de ce lemme et sachant que $\|h_p\|^2 = \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} g_{k_1, \dots, k_p}^2$, nous pouvons formuler la proposition suivante :

Proposition 3.3 Le lien entre les quantités $Q_{j,p}$ associées au noyau de Volterra h_p et $R_{j,p}$, $j = 1, 2$ associés aux coefficients de Fourier de son développement sur la base de Laguerre est le suivant :

$$R_{1,p} = \frac{p \|h_p\|^2}{1 - \xi_p^2} (2(1 + \xi_p^2) Q_{1,p} - 4\xi_p Q_{2,p} + (1 + \xi_p^2)) \quad (3.20)$$

$$R_{2,p} = \frac{2p \|h_p\|^2}{1 - \xi_p^2} (-2\xi_p Q_{1,p} + (1 + \xi_p^2) Q_{2,p} - \xi_p) \quad (3.21)$$

Preuve : voir l'annexe B.4 ■

Nous déduisons l'expression de $Q_{j,l}$ en fonction de $R_{j,l}$, $j = 1, 2$, par la résolution du système d'équations constitué par les relations (3.20) et (3.21) de la proposition précédente. On a :

$$Q_{1,p} = \frac{(1 + \xi_p^2) R_{1,p} + 2\xi_p R_{2,p}}{2(1 - \xi_p^2) p \|h_p\|^2} - \frac{1}{2} \quad (3.22)$$

$$Q_{2,p} = \frac{2\xi_p R_{1,p} + (1 + \xi_p^2) R_{2,p}}{2(1 - \xi_p^2) p \|h_p\|^2} \quad (3.23)$$

Dès lors, nous pouvons réécrire $\rho_{o,p}$ donné par (3.13) en fonction de $R_{1,p}$ et $R_{2,p}$ qui ne dépendent que du spectre de Laguerre du système¹ :

$$\rho_{o,p} = \frac{(1 + \xi_p^2) R_{1,p} + 2\xi_p R_{2,p}}{2\xi_p R_{1,p} + (1 + \xi_p^2) R_{2,p}} \quad (3.24)$$

Nous résumons ce résultat par la proposition suivante :

Proposition 3.4 *Le pôle qui caractérise la base de Laguerre optimale pour le développement du noyau de Volterra h_p est obtenu à partir des coefficients de Fourier du développement de ce même noyau sur une base de Laguerre quelconque caractérisée par le pôle ξ_p en faisant le calcul suivant :*

$$\xi_{p,opt} = \begin{cases} \rho_{o,p} - \sqrt{\rho_{o,p}^2 - 1}, & \text{si } \rho_{o,p} > 1 \\ \rho_{o,p} + \sqrt{\rho_{o,p}^2 - 1}, & \text{si } \rho_{o,p} < -1 \end{cases} \quad (3.25)$$

où $\rho_{o,p}$ est donné par l'équation (3.24).

Il est à noter que lorsque le développement du noyau de Volterra sur une base de Laguerre arbitraire, de pôle ξ_p , est infini, $\rho_{o,p}$ peut exactement être déterminé à partir de la formule (3.24).

En pratique, le développement d'un noyau de Volterra sur une base de Laguerre est tronqué à un ordre fini K . Par conséquent, pour un pôle arbitraire ξ_p , la valeur de $\rho_{o,p}$ calculée à l'aide des expressions tronquées de (3.15) et de (3.16) n'est qu'une approximation de la caractéristique réelle du système. Cette approximation dépend du pôle de Laguerre ξ_p et du spectre de Laguerre, via $R_{1,p}$ et $R_{2,p}$. Pour trouver l'approximation optimale nous proposons deux procédures itératives. La première est basée sur un traitement en bloc des données tandis que la seconde est adaptative.

¹Comme défini dans le chapitre 2, le spectre de Laguerre est l'ensemble des coefficients de Fourier.

3.3.2 Algorithme d'estimation en bloc du modèle de Volterra-Laguerre

Dans l'annexe B.4, nous avons montré que :

$$\frac{\partial R_{1,p}}{\partial \xi_p} = 2p \|h_p\|^2 \frac{\partial J_p}{\partial \xi_p}$$

En combinant cette relation avec l'équation (3.19), on obtient :

$$2p \|h_p\|^2 \frac{\partial J_p}{\partial \xi_p} = \frac{-2}{1 - \xi_p^2} R_{2,p}$$

On observe alors que lorsque J_p atteint son minimum, $R_{2,p}$ est nul. Réciproquement, le minimum de J_p étant unique pour $|\xi_p| < 1$, lorsque $R_{2,p}$ prend des valeurs proches de zéro, cela signifie que J_p est proche de son minimum et donc que le pôle ξ_p correspondant est proche de sa valeur optimale. Nous exploiterons cette propriété pour exprimer notre algorithme.

Pour des raisons pratiques évidentes, nous considérons le développement tronqué du noyau h_p sur une base à K fonctions. La procédure d'estimation est la suivante :

Algorithme 3.1 : Estimation en bloc du modèle de Volterra-Laguerre

1. Choisir un pôle stable arbitraire, construire la base de Laguerre correspondante.
2. Estimer les coefficients de Fourier optimaux au sens de l'erreur quadratique moyenne minimum ;
3. Calculer

$$T_{1,l} = \sum_{k_1=0}^{K-1} \cdots \sum_{k_p=0}^{K-1} (2k_l + 1) g_{k_1, \dots, k_p}^2, \quad l = 1, \dots, p$$

$$T_{2,l} = 2 \sum_{k_1=0}^{K-1} \cdots \sum_{k_{l-1}=0}^{K-1} \sum_{k_l=1}^{K-1} \sum_{k_{l+1}=0}^{K-1} \cdots \sum_{k_p=0}^{K-1} k_l g_{k_1, \dots, k_{l-1}, k_l, k_{l+1}, \dots, k_p} \\ \times g_{k_1, \dots, k_{l-1}, k_l-1, k_{l+1}, \dots, k_p}, \quad l = 1, \dots, p$$

4. Calculer

$$R_{1,p} = \sum_{l=1}^p T_{1,l} \quad \text{et} \quad R_{2,p} = \sum_{l=1}^p T_{2,l}$$

5. Si $R_{2,p}$ est proche de zéro, fin de l'algorithme. Sinon

- (a) Calculer $\rho_{o,p}$ à l'aide de l'équation (3.24).
- (b) Déterminer un nouveau pôle ξ_p à partir de (3.25).
- (c) Construire une nouvelle base avec ξ_p puis retour à l'étape 2.

3.3.3 Algorithme d'estimation adaptative

Le principe adopté pour cet algorithme adaptatif est de modifier le pôle de Laguerre chaque fois que les coefficients de Fourier satisfont un certain critère de convergence. Le processus d'adaptation des coefficients de Fourier se poursuit ensuite avec la nouvelle valeur estimée du pôle de Laguerre. Les coefficients de Fourier étant estimés à l'aide de l'algorithme NLMS (3.26), la procédure globale peut être vue comme étant de type bloc-NLMS.

$$G(n+1) = G(n) + \frac{\mu_{NLMS}}{a + \|S(n)\|^2} (d(n) - G(n)^T S(n)) S(n) \quad (3.26)$$

où $G(n)$ et $S(n)$ contiennent respectivement les coefficients de Fourier et les produits d'entrées filtrées associés au développement du modèle de Volterra sur des bases de Laguerre, a est une petite constante positive, μ_{NLMS} est le pas d'adaptation de l'algorithme NLMS et $d(n)$ la sortie mesurée du système.

L'estimation d'un nouveau pôle de Laguerre intervient lorsque les coefficients de Fourier deviennent relativement constants durant une fenêtre d'observation N_O donnée. La procédure globale est résumée ci-après :

Algorithme 3.2 : Estimation adaptative du modèle de Volterra-Laguerre

Paramètres :

- K : ordre de troncature.
- μ_{NLMS} : pas d'adaptation ($0 < \mu_{NLMS} < 2$).
- a : petite constante positive.
- ε_p : Seuil de convergence.
- N_O : Largeur de la fenêtre d'observation.

Initialisation :

- $G(0) = 0$ et $n = 0$.
- Choisir de manière aléatoire une valeur initiale du pôle de Laguerre ξ_p , $p = 1, \dots, P$ sur le segment $] -1, 1[$ et construire les bases correspondantes.

Calcul :

1. Estimer les spectres de Laguerre

$$G(n+1) = G(n) + \frac{\mu_{NLMS}}{a + \|S(n)\|^2} (d(n) - G^T(n) S(n)) S(n)$$

2. Pour $p = 1 \dots, P$, si $\|G_p(n+1) - G_p(n)\|^2 < \varepsilon_p$ durant l'intervalle temporel $[n+2 - N_O, n+1]$:

(a) Calculer $R_{1,p,n}$ et $R_{2,p,n}$ à l'aide de (3.17).

(b) Calculer $\rho_{o,p,n}$ à l'aide de (3.24).

(c) Déterminer la nouvelle valeur de ξ_p à l'aide de (3.25).

3. Construire la base associée, incrémenter n et retour à l'étape 1.

3.4 Modèle de Volterra-BFOR : Approche exhaustive

3.4.1 Cas du noyau de Volterra du premier ordre

Supposons que N couples d'entrée/sortie, (u, d) , du système sont disponibles et que les sorties sont bruitées. Nous allons considérer le problème d'optimisation au sens des moindres carrés de l'erreur quadratique E .

$$E = \sum_{n=0}^{N-1} (d(n) - y(n))^2 = \|d - Y\|^2 = \|d - \Phi G\|^2$$

où :

$$\begin{aligned} d &= (d(0) \ d(1) \ \cdots \ d(N-1))^T \\ Y &= (y(0) \ y(1) \ \cdots \ y(N-1))^T \\ \varphi_k &= (s_k(0) \ s_k(1) \ \cdots \ s_k(N-1))^T \\ \Phi &= (\varphi_0 \ \varphi_1 \ \cdots \ \varphi_{K-1}) \\ G &= (g_0 \ g_1 \ \cdots \ g_{K-1})^T \end{aligned} \tag{3.27}$$

φ_k et G sont des vecteurs colonne de dimension respective N et K , et Φ est une matrice de dimension $N \times K$.

Les équations normales sont regroupées sous la forme matricielle suivante :

$$\Phi^T \Phi G = \Phi^T d \tag{3.28}$$

Rappelons que les coefficients de Fourier optimaux obtenus par la résolution de (3.28) sont donnés par :

$$\hat{G} = (\Phi^T \Phi)^{-1} \Phi^T d$$

Par ailleurs, l'erreur entre la sortie du système et la sortie du modèle optimisé peut être réécrite telle que :

$$\begin{aligned} d - \hat{Y} &= d - \Phi \hat{G} \\ &= \Pi^\perp d \end{aligned} \tag{3.29}$$

où

$$\Pi^\perp = I - \Phi (\Phi^T \Phi)^{-1} \Phi^T = I - \Pi \tag{3.30}$$

Π est la matrice de projection sur le sous-espace orthogonal à celui engendré par les colonnes de la matrice Φ . En utilisant les propriétés de symétrie et d'idempotence de la matrice de projection on a :

$$(\Pi^\perp)^T (\Pi^\perp) = (\Pi^\perp)^2 = \Pi^\perp$$

Par la suite, il est possible d'écrire l'erreur quadratique minimale E_{min} en fonction de la matrice $\mathbf{\Pi}^\perp$

$$E_{min} = \|d - \hat{Y}\|^2 = d^T \mathbf{\Pi}^\perp d \quad (3.31)$$

Afin de réduire la complexité de calcul de la procédure standard des moindres carrés, nous allons recourir à la version orthogonalisée des moindres carrés [100]. En effet, la matrice des entrées filtrées, Φ , peut être factorisée en un produit d'une matrice à colonnes orthogonales avec une matrice triangulaire supérieure $\mathbf{U}$.

$$\Phi = \check{\Phi} \mathbf{U} \quad (3.32)$$

Avec $\check{\Phi} = (\check{\varphi}_0 \cdots \check{\varphi}_{K-1})$ et $\langle \check{\varphi}_k, \check{\varphi}_q \rangle = \delta_{k,q}$ où δ est le symbole de Kronecker.

En conséquence, les coefficients de Fourier optimaux sont tels que :

$$\hat{G} = \mathbf{U}^{-1} \check{\Phi}^T d \quad (3.33)$$

De même, une expression plus simple est obtenue pour l'évaluation de la matrice $\mathbf{\Pi}^\perp$:

$$\begin{aligned} \mathbf{\Pi}^\perp &= \mathbf{I} - \Phi (\Phi^T \Phi)^{-1} \Phi^T \\ &= \mathbf{I} - \check{\Phi} \mathbf{U} (\mathbf{U}^T \check{\Phi}^T \check{\Phi} \mathbf{U})^{-1} \mathbf{U}^T \check{\Phi}^T \\ &= \mathbf{I} - \check{\Phi} \mathbf{U} \mathbf{U}^{-1} (\check{\Phi}^T \check{\Phi})^{-1} \mathbf{U}^{-T} \mathbf{U}^T \check{\Phi}^T \\ &= \mathbf{I} - \check{\Phi} \check{\Phi}^T \end{aligned} \quad (3.34)$$

3.4.1.1 Procédure de sélection de pôles

Supposons qu'on dispose d'un ensemble de Q pôles réels stables $\Xi = \{\xi_0, \dots, \xi_{Q-1}\}$. Cet ensemble peut être obtenu en discrétisant le segment $] -1, 1[$. Evidemment, la dimension de cet ensemble est inversement proportionnelle au pas de discrétisation. Pour chaque ordre de développement du modèle, ces pôles sont utilisés pour générer les fonctions de base candidates. A l'ordre 1, les Q fonctions de base candidates sont :

$$F_0 = \left\{ B_m = \frac{\sqrt{1 - \xi_m^2}}{z - \xi_m}, m = 0, 1, \dots, Q-1 \right\} \quad (3.35)$$

Tandis que pour $k > 0$, les Q fonctions de bases candidates sont :

$$F_k = \left\{ B_{m,k} = \frac{\sqrt{1 - \xi_m^2}}{z - \xi_m} \prod_{i=0}^{k-1} \frac{1 - \tau_i z}{z - \tau_i} \right\} \quad (3.36)$$

où les τ_i sont les pôles sélectionnés durant les étapes précédentes.

L'idée est d'incorporer dans la base la fonction qui minimise l'erreur quadratique à chaque itération. La procédure débute avec une base vide. Elle est arrêtée lorsqu'un

certain critère est satisfait. La minimisation de ce critère permet d'optimiser la dimension de la base orthogonale. La procédure étant itérative, nous allons effectuer de façon itérative le calcul de l'erreur quadratique. Pour ce faire, il est nécessaire de déterminer une relation récursive pour l'évaluation de la matrice de projection.

Dans le cas d'une base à $(k + 1)$ fonctions, la matrice de projection s'écrit :

$$\mathbf{\Pi}_k^\perp = \mathbf{I} - \check{\mathbf{\Phi}}_k \check{\mathbf{\Phi}}_k^T$$

Incorporer une nouvelle fonction dans la base a pour effet d'introduire une colonne $\check{\varphi}_j$ dans la matrice $\check{\mathbf{\Phi}}$.

$$\check{\mathbf{\Phi}}_{k+1} = (\check{\mathbf{\Phi}}_k \quad \check{\varphi}_j)$$

En conséquence, la nouvelle matrice de projection est telle que :

$$\begin{aligned} \mathbf{\Pi}_{k+1}^\perp &= \mathbf{I} - \check{\mathbf{\Phi}}_{k+1} \check{\mathbf{\Phi}}_{k+1}^T \\ &= \mathbf{I} - \check{\mathbf{\Phi}}_k \check{\mathbf{\Phi}}_k^T - \check{\varphi}_j \check{\varphi}_j^T \\ &= \mathbf{\Pi}_k^\perp - \check{\varphi}_j \check{\varphi}_j^T \end{aligned} \quad (3.37)$$

Par suite, l'incrément de l'erreur quadratique minimale, $\Delta E_{k+1} = E_k - E_{k+1}$, s'écrit :

$$\Delta E_{k+1} = d^T \left(\mathbf{\Pi}_k^\perp - \mathbf{\Pi}_{k+1}^\perp \right) d = (d^T \check{\varphi}_j)^2 \quad (3.38)$$

3.4.1.2 Critère d'arrêt

Différents critères utilisés pour la sélection de modèles peuvent servir comme critère d'arrêt de l'algorithme [57, 100, 51]. On citera par exemple le critère d'information d'Akaike (AIC) et le critère BIC (Bayesian Information Criterion) ou MDL (Minimum Description Length). Ces deux critères sont respectivement définis par :

$$AIC(k) = \log \left(\frac{E_k}{N} \right) + \frac{2}{N} k \quad (3.39)$$

$$BIC(k) = \log \left(\frac{E_k}{N} \right) + \frac{\log(N)}{N} k \quad (3.40)$$

En résumé l'algorithme proposé est le suivant [65]

Algorithme 3.3 Sélection des pôles d'une base BFOR (Cas linéaire)

1. Discrétisation du segment $] -1, 1[$ avec un pas fixe pour obtenir un ensemble de pôles candidats stables.
2. Sélection de la première fonction de la base :
 - (a) Construction de l'ensemble F_0 de fonctions candidates.
 - (b) Formation du vecteur φ_0 en calculant les entrées filtrées $s_0(n)$, $n = 0, 1, \dots, N - 1$, pour chaque fonction candidate $B_m(z)$.

- (c) Normalisation de $\Phi_1 = (\varphi_0)$.
Sélection du pôle ξ_m de la fonction $B_m(z)$ qui minimise l'erreur quadratique.
- 3. Construction itérative de la base optimale pour chaque $(k+1)$ -è itération, $k > 0$:
 - (a) Construction de l'ensemble F_k de fonctions candidates
 - (b) Formation du vecteur φ_k en calculant les entrées filtrées $s_k(n)$, $n = 0, 1, \dots, N-1$, pour chaque fonction candidate $B_{m,k}(z)$.
 - (c) Orthonormalisation du nouveau vecteur colonne φ_k pour former la matrice Φ_{k+1} et sélection du pôle ξ_k de la composante passe-bas de la fonction $B_{m,k}(z)$ qui maximise l'incrément d'erreur quadratique (3.38).
 - (d) Calcul de l'ordre de troncature au moyen du critère d'arrêt (3.39) ou (3.40).
- 4. Si le critère AIC ou BIC est minimal, les coefficients de Fourier sont estimés et la procédure s'arrête. Sinon retour à l'étape (3a) en remplaçant k par $k+1$.

3.4.2 Cas général

La procédure formulée ci-dessus peut-être étendue au cas plus général d'un modèle de Volterra d'ordre quelconque [65, 68, 66]. Considérons un modèle de Volterra d'ordre P dont le développement tronqué de chacun des noyaux est :

$$h_p(n_1, \dots, n_p) = \sum_{k_1=0}^{K_p-1} \cdots \sum_{k_p=0}^{K_p-1} g_{k_1, \dots, k_p} b_{k_1,p}(n_1) \cdots b_{k_p,p}(n_p), \quad p = 1, \dots, P$$

où les fonctions de base $b_{k,p}$ sont définies par [90] :

$$b_{0,p}(n) = \mathcal{Z}^{-1} \left\{ B_{0,p}(z) = \frac{\sqrt{1 - \xi_{0,p}^2}}{z - \xi_{0,p}} \right\}$$

$$b_{k,p}(n) = \mathcal{Z}^{-1} \left\{ B_{k,p}(z) = \frac{\sqrt{1 - \xi_{k,p}^2}}{z - \xi_{k,p}} \prod_{i=0}^{k-1} \frac{1 - \xi_{i,p} z}{z - \xi_{i,p}} \right\}, \quad k = 1, \dots, K_p - 1$$

Rappelons que la sortie globale du modèle est la somme des sorties de chaque bloc homogène : $y(n) = \sum_{p=1}^P y_p(n)$ où la sortie y_p est donnée par :

$$y_p(n) = \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=0}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n - n_j)$$

Rappelons aussi que le modèle résultant après décomposition sur une BFOR, donné par (3.4), est :

$$y_p(n) = \sum_{k_1=0}^{K_p-1} \cdots \sum_{k_p=k_{p-1}}^{K_p-1} g_{k_1, \dots, k_p} s_{k_1, \dots, k_p}(n)$$

3.4. Modèle de Volterra-BFOR : Approche exhaustive

où : $s_{k_1, \dots, k_p}(n) = \prod_{j=1}^p s_{k_j, p}(n)$, avec $s_{k_j, p}(n) = \sum_{i=0}^{M-1} b_{k_j, p}(i) u(n-i)$. Les coefficients $g_{k_1, \dots, k_p}$ sont obtenus en appliquant une procédure de triangularisation, similaire à celle décrite pour les noyaux de Volterra dans le chapitre 1, à l'ensemble des coefficients $\dot{g}_{k_1, \dots, k_p}$.

La complexité paramétrique du modèle global est donc égale à :

$$C^{BFOR}(K_1, \dots, K_p) = \sum_{p=1}^P \frac{(K_p + p - 1)!}{(K_p - 1)! p!} \quad (3.41)$$

Supposons à présent que nous disposons de N mesures d'entrée-sortie. Nous définissons :

- les vecteurs $\varphi_{k_1, \dots, k_p}$ des produits à l'ordre p , $p = 1, \dots, P$ des entrées filtrées $s_{k_i, p}(n)$, $i = 1, \dots, p$:

$$\varphi_{k_1, \dots, k_p} = (s_{k_1, \dots, k_p}(0), \dots, s_{k_1, \dots, k_p}(N-1))^T$$

- pour $p \geq 2$ les matrices $\mathbf{D}_{q, p}$, $q = 0, 1, \dots, K_p - 1$:

$$\mathbf{D}_{q, p} = \left(\varphi_{0, 0, \dots, 0, q} \cdots \varphi_{k_1, k_2, \dots, k_{p-1}, q} \cdots \varphi_{q, q, \dots, q, q} \right)$$

avec $0 \leq k_1 \leq k_2 \leq \dots \leq k_{p-1} \leq q$.

Chaque matrice $\mathbf{D}_{q, p}$ est de dimensions $N \times \frac{(q+p)!}{q! p!}$.

- les matrices :

$$\begin{aligned} \Phi_1 &= (\varphi_0 \cdots \varphi_{K_1-1}) \\ \Phi_p &= (\mathbf{D}_{0, p} \cdots \mathbf{D}_{q, p} \cdots \mathbf{D}_{K_p-1, p}), \quad p \geq 2 \\ \Phi &= (\Phi_1 \cdots \Phi_p \cdots \Phi_P) \end{aligned}$$

Chaque matrice Φ_p a une structure en bloc. Elle est exactement constituée de K_p blocs de type $\mathbf{D}_{q, p}$. La matrice Φ_p est de dimensions $N \times \frac{(K_p-1+p)!}{(K_p-1)! p!}$.

- les vecteurs des coefficients de Fourier :

$$\begin{aligned} G_p &= (g_{0, \dots, 0, 0} \cdots g_{0, \dots, 0, k_p} g_{0, \dots, 1, k_p} \cdots g_{k_1, \dots, k_p} \cdots g_{0, \dots, 0, K_p-1} \cdots \\ &\quad \cdots g_{K_p-1, \dots, K_p-1, K_p-1})^T \\ G &= (G_1^T \cdots G_p^T \cdots G_P^T)^T \end{aligned}$$

Exemple 3.1 Pour illustrer ces définitions, prenons le cas d'un modèle quadratique, dont les noyaux sont développés sur une BFOR à 4 fonctions. On a :

–

$$\begin{aligned} \varphi_{k_1} &= (s_{k_1}(0), \dots, s_{k_1}(N-1))^T, \quad k_1 = 0, 1, 2, 3 \\ \varphi_{k_1, k_2} &= (s_{k_1, k_2}(0), \dots, s_{k_1, k_2}(N-1))^T, \quad k_1 = 0, 1, 2, 3, \quad k_2 = k_1, \dots, 3 \end{aligned}$$

– pour le noyau quadratique nous avons les matrices $\mathbf{D}_{q,2}$, $q = 0, 1, 2, 3$:

$$\begin{aligned}\mathbf{D}_{0,2} &= (\varphi_{0,0}) \\ \mathbf{D}_{1,2} &= (\varphi_{0,1} \quad \varphi_{1,1}) \\ \mathbf{D}_{2,2} &= (\varphi_{0,2} \quad \varphi_{1,2} \quad \varphi_{2,2}) \\ \mathbf{D}_{3,2} &= (\varphi_{0,3} \quad \varphi_{1,3} \quad \varphi_{2,3} \quad \varphi_{3,3})\end{aligned}$$

– on définit les matrices :

$$\begin{aligned}\Phi_1 &= (\varphi_0 \quad \varphi_1 \quad \varphi_2 \quad \varphi_3) \\ \Phi_2 &= (\mathbf{D}_{0,2} \quad \mathbf{D}_{1,2} \quad \mathbf{D}_{2,2} \quad \mathbf{D}_{3,2}) \\ \Phi &= (\Phi_1 \quad \Phi_2)\end{aligned}$$

La matrice Φ_2 est partitionnée en 4 blocs $\mathbf{D}_{q,p}$, $q = 0, 1, 2, 3$.

– les vecteurs des coefficients de Fourier sont :

$$\begin{aligned}G_1 &= (g_0 \quad g_1 \quad g_2 \quad g_3)^T \\ G_2 &= (g_{0,0} \quad g_{0,1} \quad g_{1,1} \quad g_{0,2} \quad g_{1,2} \quad g_{2,2} \quad g_{0,3} \quad g_{1,3} \quad g_{2,3} \quad g_{3,3})^T \\ G &= (G_1^T \quad G_2^T)^T\end{aligned}$$

Par suite, le vecteur $Y = (y(0) \ y(1) \cdots y(N-1))^T$ des sorties du modèle de Volterra durant l'intervalle temporel $[0, N-1]$ peut s'écrire :

$$Y = \Phi G$$

Nous pouvons dès lors formuler le problème d'optimisation des coefficients de Fourier comme suit :

$$\hat{G} = \arg \min_G \|d - \Phi G\|^2$$

ce qui conduit à la solution bien connue des moindres carrés suivante :

$$\hat{G} = (\Phi^T \Phi)^{-1} \Phi^T d$$

où $d = (d(0) \ d(1) \cdots d(N-1))^T$ est le vecteur des sorties réelles du système. En définissant la matrice de projection orthogonale $\Pi^\perp$ sur le sous-espace orthogonal à celui engendré par les colonnes de Φ , par :

$$\Pi^\perp = I - \Phi (\Phi^T \Phi)^{-1} \Phi^T = I - \Pi$$

le vecteur des erreurs de prédiction s'écrit donc :

$$d - \hat{Y} = d - \Phi \hat{G} = \Pi^\perp d$$

Comme nous l'avons déjà montré dans le cas linéaire, tenant compte des propriétés de la matrice de projection, l'erreur quadratique minimale s'écrit :

$$E_{min} = \|d - \hat{Y}\|^2 = d^T \Pi^\perp d \quad (3.42)$$

Appliquons à présent à la matrice Φ une factorisation de type QR. On a : $\Phi = \check{\Phi}U$, où $\check{\Phi}$ est une matrice à colonnes orthonormales et U une matrice triangulaire supérieure. Les coefficients de Fourier optimaux s'obtiennent alors grâce à la relation suivante :

$$\hat{G} = U^{-1}\check{\Phi}^T d \quad (3.43)$$

Il est à noter que la matrice $\check{\Phi}$ hérite de la structure en bloc de Φ . En conséquence, nous avons :

$$\begin{aligned} \check{\Phi} &= (\check{\Phi}_1 \cdots \check{\Phi}_p \cdots \check{\Phi}_P) \\ \check{\Phi}_p &= (\check{D}_{0,p} \cdots \check{D}_{q,p} \cdots \check{D}_{K_p-1,p}) \end{aligned}$$

Où chaque bloc $\check{D}_{q,p}$ est formé par les colonnes $\check{\phi}_{k_1, \dots, k_{p-1}, q}$, $0 \leq k_1 \leq \dots \leq k_{p-1} \leq q$. Les relations d'orthogonalité suivantes sont vérifiées :

$$\begin{aligned} \check{\Phi}_{p_1}^T \check{\Phi}_{p_2} &= \delta_{p_1, p_2} \mathbf{I} \\ \check{D}_{q_1, p}^T \check{D}_{q_2, p} &= \delta_{q_1, q_2} \mathbf{I} \end{aligned}$$

où δ est le symbole de Kronecker et $\mathbf{I}$ la matrice identité.

Définissons à présent la matrice Π_p comme étant la matrice de projection orthogonale sur le sous-espace engendré par les colonnes de $\check{\Phi}_p$:

$$\Pi_p = \check{\Phi}_p \check{\Phi}_p^T$$

Nous pouvons alors formuler la proposition suivante :

Proposition 3.5 *La matrice de projection orthogonale sur le sous-espace engendré par les colonnes de la matrice Φ est la somme des matrices de projection sur les sous-espaces engendrés par les colonnes des matrices orthogonales $\check{\Phi}_p$, $p = 1, \dots, P$:*

$$\Pi = \sum_{p=1}^P \Pi_p \quad (3.44)$$

Preuve : La preuve de cette proposition découle d'un calcul direct :

$$\begin{aligned} \Pi &= \Phi (\Phi^T \Phi)^{-1} \Phi^T \\ &= \check{\Phi} U U^{-1} (\check{\Phi}^T \check{\Phi})^{-1} U^{-T} U^T \check{\Phi}^T \\ &= \check{\Phi} \check{\Phi}^T \end{aligned}$$

Tenant compte de la structure en bloc de la matrice $\check{\Phi}$, il vient que :

$$\Pi = \sum_{p=1}^P \check{\Phi}_p \check{\Phi}_p^T = \sum_{p=1}^P \Pi_p$$

■

Définissons à présent les erreurs quadratiques de prédiction suivantes :

$$E_{min,p} = \|d - \hat{Y}_p\|^2 = d^T \mathbf{\Pi}_p^\perp d, \quad p = 1, \dots, P$$

où $\hat{Y}_p = \check{\Phi}_p \hat{G}_p$

Nous avons alors le corollaire suivant :

Corollaire 3.1 *L'erreur quadratique de prédiction minimale E_{min} se décompose en la somme d'erreurs quadratiques de prédiction $E_{min,p}$ relatives à chaque noyau tel que :*

$$E_{min} = \sum_{p=1}^P E_{min,p} - (P-1)d^T d \quad (3.45)$$

Preuve : Calculons la différence $E_{min} - \sum_{p=1}^P E_{min,p}$, en utilisant la proposition 3.5. On a :

$$\begin{aligned} E_{min} - \sum_{p=1}^P E_{min,p} &= d^T \mathbf{\Pi}^\perp d - \sum_{p=1}^P d^T \mathbf{\Pi}_p^\perp d \\ &= d^T d - \sum_{p=1}^P d^T \mathbf{\Pi}_p d - P d^T d + \sum_{p=1}^P d^T \mathbf{\Pi}_p d \\ &= -(P-1)d^T d \end{aligned}$$

d'où le résultat. ■

3.4.2.1 Approche constructive pour la sélection des pôles

Notre objectif est de construire de manière itérative une base $\mathfrak{B}_p$ pour le développement de chaque noyau h_p . Afin de choisir les pôles de la BFOR, nous supposons que nous disposons, en plus des N mesures d'entrée-sortie $\{u(n), d(n)\}_{n=0}^{N-1}$, d'un ensemble de R pôles réels stables $\Xi = \{\xi_0, \dots, \xi_{R-1}\}$. En l'absence d'information *a priori*, cet ensemble de pôles peut être obtenu en discrétisant avec un pas donné le segment $] -1, 1[$. Evidemment, la complexité de calcul de l'algorithme dépend du pas de discrétisation. Comme dans le cas linéaire, l'ensemble des fonctions candidates est donné, pour chaque noyau h_p , respectivement pour la première itération et pour les itérations suivantes, par :

$$\begin{aligned} F_{0,p} &= \left\{ \mathcal{Z}^{-1} \left\{ B_{m,p}(z) = \frac{\sqrt{1 - \xi_{m,p}^2}}{z - \xi_{m,p}} \right\}, 0 \leq m \leq R-1 \right\} \\ F_{k,p} &= \left\{ \mathcal{Z}^{-1} \left\{ B_{m,k,p}(z) = \frac{\sqrt{1 - \xi_{m,p}^2}}{z - \xi_{m,p}} \prod_{i=0}^{k-1} \frac{1 - \tau_{i,p} z}{z - \tau_{i,p}} \right\}, 0 \leq m \leq R-1 \right\}, \quad k > 0 \end{aligned}$$

Les pôles $\tau_{i,p}, i \in \{0, \dots, k-1\}$, correspondent à ceux déterminés dans les étapes précédentes de l'algorithme. Nous signalons qu'en raison du caractère itératif de la construction de la base, nous ajoutons aux différentes quantités (matrices, vecteurs et scalaires) jusque là définies, un indice supplémentaire $k \geq 0$ qui indique la $(k+1)$ -ème itération et donc la $(k+1)$ -ème fonction de la base.

$\mathfrak{B}_{k,p}$ est une base à k fonctions associée au développement du noyau de Volterra d'ordre p . Les entrées filtrées, ou leurs produits, par ses fonctions sont rangées dans la matrice $\check{\Phi}_{k,p}$.

Ajoutons une nouvelle fonction à $\mathfrak{B}_{k,p}$. Nous obtenons la base $\mathfrak{B}_{k+1,p}$ à laquelle nous associons la matrice $\check{\Phi}_{k+1,p}$. Nous pouvons vérifier que $\check{\Phi}_{k+1,p}$ est obtenue à partir de $\check{\Phi}_{k,p}$ en lui ajoutant un bloc $\check{\mathbf{D}}_{k,p}$. En effet :

$$\begin{aligned}\check{\Phi}_{k+1,p} &= (\check{\mathbf{D}}_{0,p} \cdots \check{\mathbf{D}}_{k-1,p} \quad \check{\mathbf{D}}_{k,p}) \\ &= \begin{pmatrix} \check{\Phi}_{k,p} & \check{\mathbf{D}}_{k,p} \end{pmatrix}\end{aligned}\tag{3.46}$$

Proposition 3.6 *Lorsqu'une nouvelle fonction est introduite dans la base $\mathfrak{B}_{k,p}$, l'erreur quadratique de prédiction associée au noyau d'ordre p est réduite.*

Preuve : Tenant compte de (3.46), nous pouvons déduire une relation récursive pour la matrice de projection $\Pi_p^\perp$. On a :

$$\begin{aligned}\Pi_{k+1,p}^\perp &= \mathbf{I} - \check{\Phi}_{k+1,p} \check{\Phi}_{k+1,p}^T \\ &= \mathbf{I} - \check{\Phi}_{k,p} \check{\Phi}_{k,p}^T - \check{\mathbf{D}}_{k,p} \check{\mathbf{D}}_{k,p}^T \\ &= \Pi_{k,p}^\perp - \check{\mathbf{D}}_{k,p} \check{\mathbf{D}}_{k,p}^T\end{aligned}$$

En utilisant l'expression qui lie l'erreur quadratique de prédiction à la matrice de projection, nous obtenons :

$$\begin{aligned}\Delta E_{min,k+1,p} &= E_{min,k,p} - E_{min,k+1,p} \\ &= d^T (\Pi_{k,p}^\perp - \Pi_{k+1,p}^\perp) d \\ &= d^T \check{\mathbf{D}}_{k,p} \check{\mathbf{D}}_{k,p}^T d \\ &= \sum_{k_1=0}^k \sum_{k_2=k_1}^k \cdots \sum_{k_{p-1}=k_{p-2}}^k \left(d^T \check{\phi}_{k_1,k_2,\dots,k_{p-1},k} \right)^2 \geq 0\end{aligned}$$

Ce qui implique que $E_{min,k+1,p} \leq E_{min,k,p}$, i.e. la réduction de l'erreur quadratique de prédiction. ■

Corollaire 3.2 *L'incrément d'erreur quadratique de prédiction, dû à l'introduction d'une nouvelle fonction dans les bases respectives associées à chacun des noyaux, est la somme des incréments d'erreurs quadratiques de prédiction associées à chaque noyau.*

$$\Delta E_{min,k} = \sum_{p=1}^P \Delta E_{min,k,p}\tag{3.47}$$

Preuve :

$$\begin{aligned}
 \Delta E_{min,k} &= E_{min,k-1} - E_{min,k} \\
 &= \sum_{p=1}^P E_{min,k-1,p} - (P-1)d^T d - \sum_{p=1}^P E_{min,k,p} + (P-1)d^T d \\
 &= \sum_{p=1}^P (E_{min,k-1,p} - E_{min,k,p}) = \sum_{p=1}^P \Delta E_{min,k,p}
 \end{aligned}$$

■

Nous proposons de choisir les pôles des différentes bases de sorte que l'incrément d'erreur quadratique de prédiction soit maximisé ce qui équivaut à réduire l'erreur quadratique de prédiction. Ce problème d'optimisation se formule donc ainsi :

$$(\tau_{k,1}, \dots, \tau_{k,p}, \dots, \tau_{k,P}) = \arg \max_{\xi_{m,1}, \dots, \xi_{m,p}, \dots, \xi_{m,P}} \Delta E_{min,k+1}(\xi_{m,1}, \dots, \xi_{m,p}, \dots, \xi_{m,P}) \quad (3.48)$$

A l'instar du développement réalisé dans le cas linéaire, la procédure de sélection est arrêtée en utilisant un critère de sélection de modèle comme AIC ou BIC.

Deux stratégies sont possibles pour maximiser l'incrément d'erreur de prédiction. Soit maximiser directement $\Delta E_{min,k+1}$ soit maximiser séparément les $\Delta E_{min,k+1,p}$. La première stratégie est indiquée lorsqu'une seule base est utilisée pour le développement de tous les noyaux tandis que la seconde l'est dans le cas où des bases distinctes sont construites pour chaque noyau.

Selon que l'on effectue le choix d'une base unique ou de différentes bases, nous avons les algorithmes ci-après :

Algorithme 3.4 : Choix d'une base unique

1. Discrétisation du segment $] -1, 1[$ avec un pas fixé pour obtenir les pôles candidats.
2. Sélection de la première fonction de base :
 - (a) Construire l'ensemble F_0 des fonctions candidates.
 - (b) Former les vecteurs $\phi_{0,\dots,0}$ pour chaque ordre de non-linéarité p en calculant les entrées filtrées pour chaque fonction candidate $B_m(z)$.
 - (c) Construire et orthonormaliser la matrice Φ_1 , et choisir le premier pôle tel que l'erreur quadratique de prédiction globale soit minimale.
3. A la $(k+1)$ -ème itération, $k > 0$:
 - (a) Former l'ensemble F_k de fonctions candidates.
 - (b) Former les vecteurs $\phi_{k_1,k_2,\dots,k_{p-1},k}$, $k_{i+1} = k_i, \dots, k$, $i = 1, \dots, p-2$ en calculant les entrées filtrées et leurs produits pour chaque fonction candidate $B_{m,k}(z)$, $m = 0, \dots, R-1$.

- (c) Orthonormalisation des nouveaux vecteurs colonnes et formation de la matrice $\check{\Phi}_{k+1}$ puis sélection du pôle ξ_k de la composante passe-bas de la fonction $B_{m,k}(z)$ qui maximise l'incrément d'erreur quadratique de prédiction.
- (d) Evaluer le critère d'arrêt.
- 4. Si le critère est minimal, estimer les coefficients de Fourier et arrêter la procédure. Sinon, retour à l'étape 3a en incrémentant k .

Algorithme 3.5 : Choix d'une base différente pour chaque noyau

1. Discrétisation du segment $] -1, 1[$ avec un pas fixé pour obtenir les pôles candidats.
2. Construire l'ensemble F_0 des fonctions candidates.
3. Sélection des premières fonctions de chaque base $\mathfrak{B}_p$. Pour $p = 1, \dots, P$:
 - (a) Former les vecteurs $\phi_{0,\dots,0}$ en calculant les entrées filtrées $s_{0,p}$ et en faisant leur produit d'ordre p , pour chaque fonction candidate.
 - (b) Orthonormaliser $\phi_{0,\dots,0}$ pour constituer $\check{\Phi}_{1,p}$ et donc $\check{\Phi}$ puis choisir le pôle $\xi_{m,p}$ de la fonction $B_{m,p}(z)$ qui minimise $E_{min,1,p}$.
4. A la $(k+1)$ -ème itération, $k > 0$, pour $p = 1, \dots, P$:
 - (a) Former l'ensemble $F_{k,p}$ des fonctions candidates.
 - (b) Former les vecteurs $\phi_{k_1,k_2,\dots,k_{p-1},k}$ en calculant les entrées filtrées $s_{k,p}$ et en faisant leur produit d'ordre p , pour chaque fonction candidate.
 - (c) Orthonormaliser $\phi_{k_1,k_2,\dots,k_{p-1},k}$ pour constituer $\check{\Phi}_{k+1,p}$ et donc $\check{\Phi}_{k+1}$ puis choisir le pôle $\xi_{m,p}$ de la fonction $B_{m,k,p}(z)$ qui maximise $\Delta E_{min,k+1,p}$.
5. Evaluer le critère d'arrêt.
6. Si ce critère est minimal, estimer les coefficients de Fourier et arrêter la procédure. Sinon, retour à l'étape 4a en incrémentant k .

3.4.2.2 Procédure de seuillage

Dans la procédure décrite ci-dessus, le critère d'arrêt donne un ordre de troncature optimal K au sens du critère utilisé. A cet ordre de troncature correspond un nombre de paramètres du modèle égal à $C^{BFOR}(K, \dots, K)$, donné par (3.41). L'erreur quadratique moyenne normalisée correspondante est alors :

$$EQMN \left(C^{BFOR}(K, \dots, K) \right) = 10 \log \frac{\sum_{n=0}^{N-1} (d(n) - y(n))^2}{\sum_{n=0}^{N-1} (d(n))^2} \quad (3.49)$$

Il est vrai que la procédure que nous venons de présenter permet d'obtenir une représentation parcimonieuse du modèle de Volterra. Cependant, l'ordre de troncature utilisé

étant le même pour tous les noyaux, il est possible que certains termes non significatifs soient présents dans le modèle. Nous proposons de les écarter en "élaguant" le modèle de manière rétrograde. Une telle procédure réduit plus significativement la complexité du modèle. En contrepartie, le modèle se dégrade. Cette dégradation n'est pas considérable si le terme retranché du modèle est peu significatif. Nous allons donc supprimer les paramètres un à un, tout en s'assurant que la dégradation du modèle soit bornée par un seuil choisi *a priori*.

On définit par $EQMN(C^{BFOR}(K_1, \dots, K_p))$ l'erreur quadratique moyenne normalisée correspondant à un ordre de troncature K_p pour la base $\mathfrak{B}_p$ et par $\lambda(\cdot)$ la différence suivante :

$$\lambda(K_1, \dots, K_p) = EQMN(C^{BFOR}(K_1, \dots, K_p)) - EQMN(C^{BFOR}(K, \dots, K)) \quad (3.50)$$

Désignons par K_p^s l'ordre de troncature après seuillage de la base $\mathfrak{B}_p$ tel que :

$$(K_1^s, \dots, K_p^s) = \arg \max_{K_1, \dots, K_p} \lambda(K_1, \dots, K_p) \quad (3.51)$$

sous la contrainte suivante :

$$0 \leq \lambda(K_1, \dots, K_p) \leq \frac{|EQN(C^{BFOR}(K, K))|}{\kappa} \quad (3.52)$$

avec κ une constante positive choisie de sorte que la dégradation de l'EQMN soit limitée malgré le seuillage de certains paramètres :

$$C(K_1^s, \dots, K_p^s) \leq C^{BFOR}(K, \dots, K)$$

Il est à noter que la complexité paramétrique d'un noyau donné est d'autant plus élevée que l'ordre de non-linéarité qui lui est associé est important. Ainsi, la modification de la complexité relative à un noyau d'ordre donné a d'autant plus d'impact sur la complexité globale du modèle que l'ordre de non-linéarité du noyau est élevé. Nous proposons donc de débiter le seuillage au niveau du noyau de degré le plus élevé. Puis nous décrétons ce degré. La procédure se résume comme suit :

Algorithme 3.6 : Procédure de seuillage

Pour $p = P, P-1, \dots, 1$:

1. Réduire K_p et évaluer $\lambda(K, \dots, K, K_p, K_{p+1}^s, \dots, K_p^s)$
2. Si la contrainte (3.52) n'est plus vérifiée, arrêter. On a donc K_p^s . Puis décrétement p . Sinon retour à 1.

3.4.3 Modèle de Volterra-BFOR : Approche analytique

Le principal écueil de l'approche que nous venons de présenter est d'évaluer la fonction de coût à optimiser de manière systématique pour chaque élément de l'espace de

recherche. En l'absence d'information *a priori* sur le système, cet espace est obtenu par discrétisation du segment $] -1, 1[$. Plus, la recherche se veut être fine, i.e. un pas de discrétisation très faible, plus la méthode est lourde à mettre en oeuvre. Plus la recherche est grossière, i.e. un pas de discrétisation important, moins on est sûr de trouver la bonne solution. Dans ce type de problèmes d'optimisation, un recours aux techniques d'optimisation globale comme les algorithmes génétiques, le recuit simulé ou l'algorithme des colonies de fourmis peut être envisagé. Cependant, ces techniques bien qu'étant efficaces requièrent généralement un nombre important d'évaluations de la fonction de coût. Ce qui rejoint l'écueil principal de la méthode de recherche exhaustive que nous avons présentée. Dans cette dernière, l'avantage est qu'à l'aide du pas de discrétisation, on fixe *a priori* le nombre d'évaluations de la fonction de coût à chaque itération.

Nous retiendrons toutefois que le principe de la méthode est attrayant. Peut-on utiliser une autre technique pour le choix du pôle optimal à chaque itération, qui soit moins coûteuse ?

Nous avons déjà indiqué que l'avantage d'un développement sur des bases de Laguerre est qu'il est possible de déterminer les pôles optimaux de manière analytique. La méthode que nous avons présentée dans la section 3.3 présente l'avantage d'être indépendante vis-à-vis de l'ordre de troncature. Nous nous proposons donc de l'associer à l'approche constructive de la section 3.4.2.

Rappelons que le vecteur Y des sorties du modèle s'écrit :

$$Y = \Phi G = \check{\Phi} \check{G}$$

où $\Phi = \check{\Phi}U$ et $\check{G} = UG$. Rappelons aussi que $\check{\Phi}$ a une structure en bloc qui évolue avec l'augmentation de l'ordre de troncature. L'orthogonalité de $\check{\Phi}$ permet d'avoir une estimation découplée du modèle. En vue de fixer les idées, prenons l'exemple d'un modèle linéaire quadratique développé sur des bases à 2 fonctions. La matrice $\check{\Phi}$ s'écrit :

$$\check{\Phi} = \begin{pmatrix} \check{s}_0(0) & \check{s}_1(0) & \check{s}_{0,0}(0) & \check{s}_{0,1}(0) & \check{s}_{1,1}(0) \\ \check{s}_0(1) & \check{s}_1(1) & \check{s}_{0,0}(1) & \check{s}_{0,1}(1) & \check{s}_{1,1}(1) \\ \vdots & & \vdots & & \vdots \\ \check{s}_0(N-1) & \check{s}_1(N-1) & \check{s}_{0,0}(N-1) & \check{s}_{0,1}(N-1) & \check{s}_{1,1}(N-1) \end{pmatrix}$$

Nous allons à présent réorganiser les colonnes de $\check{\Phi}$ de sorte à grouper d'une part les colonnes qui ne dépendent que de la première fonction de base et de l'autre celles qui dépendent à la fois de la première et de la seconde fonctions. Une telle organisation s'obtient en permutant la deuxième et la troisième colonne de $\check{\Phi}$. Nous obtenons :

$$\bar{\Phi} = \begin{pmatrix} \check{s}_0(0) & \check{s}_{0,0}(0) & \check{s}_1(0) & \check{s}_{0,1}(0) & \check{s}_{1,1}(0) \\ \check{s}_0(1) & \check{s}_{0,0}(1) & \check{s}_1(1) & \check{s}_{0,1}(1) & \check{s}_{1,1}(1) \\ \vdots & & \vdots & & \vdots \\ \check{s}_0(N-1) & \check{s}_{0,0}(N-1) & \check{s}_1(N-1) & \check{s}_{0,1}(N-1) & \check{s}_{1,1}(N-1) \end{pmatrix}$$

D'où la structure en bloc suivante :

$$\bar{\Phi} = \begin{pmatrix} \bar{\Phi}_0 & \bar{\Phi}_1 \end{pmatrix}$$

Avec :

$$\bar{\Phi}_0 = \begin{pmatrix} \check{s}_0(0) & \check{s}_{0,0}(0) \\ \check{s}_0(1) & \check{s}_{0,0}(1) \\ \vdots & \vdots \\ \check{s}_0(N-1) & \check{s}_{0,0}(N-1) \end{pmatrix}, \bar{\Phi}_1 = \begin{pmatrix} \check{s}_1(0) & \check{s}_{0,1}(0) & \check{s}_{1,1}(0) \\ \check{s}_1(1) & \check{s}_{0,1}(1) & \check{s}_{1,1}(1) \\ \vdots & \vdots & \vdots \\ \check{s}_1(N-1) & \check{s}_{0,1}(N-1) & \check{s}_{1,1}(N-1) \end{pmatrix}$$

Cette permutation se traduit par une permutation de la deuxième et de la troisième ligne du vecteur des coefficients de Fourier "orthogonalisés", $\bar{G}$. On a :

$$\bar{G} = \begin{pmatrix} \check{g}_0 & \check{g}_{0,0} & \check{g}_1 & \check{g}_{1,1} & \check{g}_{0,1} \end{pmatrix}^T = \begin{pmatrix} \bar{G}_0^T & \bar{G}_1^T \end{pmatrix}^T$$

D'une manière générale, nous définissons :

- $\bar{\Phi}_k$: matrice dont les colonnes sont $\check{\phi}_k$ et $\check{\phi}_{k_1, \dots, k_{p-1}, k}$ pour $0 \leq k_1 \leq \dots \leq k_{p-1} \leq k$ et $2 \leq p \leq P$.
- $\bar{G}_k$: vecteur dont les composantes sont $\check{g}_k$ et $\check{g}_{k_1, \dots, k_{p-1}, k}$ pour $0 \leq k_1 \leq \dots \leq k_{p-1} \leq k$ et $2 \leq p \leq P$.

Avec ces définitions, la sortie du modèle s'écrit donc de manière équivalente comme :

$$Y = \bar{\Phi} \bar{G} = \sum_{k=0}^{K-1} \bar{\Phi}_k \bar{G}_k$$

Par suite :

$$y(n) = \sum_{k=0}^{K-1} \bar{y}_k(n), \quad \bar{y}_k(n) = \bar{S}_k(n) \bar{G}_k \quad (3.53)$$

$\bar{S}_k(n)$ étant la $(n+1)$ -ième ligne de $\bar{\Phi}_k$. Par suite, nous avons la structure multicanal de la figure 3.2.

Nous définissons les signaux suivants ($k \geq 0$) :

- $\hat{y}_k(n)$: la sortie cumulée jusqu'à la $(k+1)$ -ième branche :

$$\hat{y}_k(n) = \sum_{i=0}^k \bar{y}_i(n)$$

- $v_{k+1}(n)$: le signal résiduel donné par :

$$v_{k+1}(n) = d(n) - \hat{y}_k(n)$$

où $d(n)$ est la sortie réelle du système et $v_0(n) = d(n)$.

Nous pouvons vérifier que les signaux résiduels v_k obéissent à la relation de récurrence suivante :

$$v_{k+1}(n) = v_k(n) - \bar{y}_k(n) \quad (3.54)$$

Le principe de la méthode que nous proposons rejoint celui couramment utilisé pour l'identification des modèles parallèle-cascade [102, 71, 137]. Il s'agit d'identifier à chaque itération, $k \geq 0$, un modèle entre l'entrée u et le signal résiduel v_k . Nous proposons l'algorithme suivant :

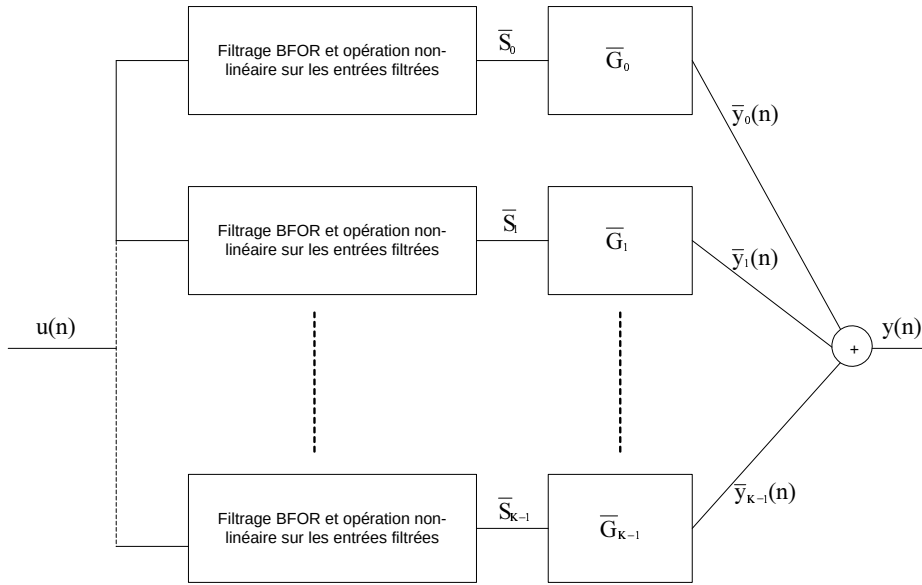


FIG. 3.2 – Représentation multicanal équivalente du modèle de Volterra-BFOR orthogonalisé

Algorithme 3.7 : Approche analytique

1. Pour $k = 0, \dots, K - 1$:

- (a) Déterminer les pôles optimaux de Laguerre $\xi_{k,p}$, $p = 1, \dots, P$ en utilisant comme entrées-sorties les signaux u et v_k , et en appliquant la méthode décrite par l'algorithme 3.1.
- (b) Construire les fonctions $b_{k,p}$ avec les pôles déterminés.
- (c) Faire passer l'entrée par les filtres de la BFOR. Après orthonormalisation, générer les vecteurs $S_k(n)$ puis former la matrice Φ_k .
- (d) Estimer le vecteur $\bar{G}_k$:

$$\bar{G}_k = \bar{\Phi}_k^T v_k$$

où v_k est le vecteur des signaux résiduels $v_k(n)$ $n = 0, \dots, N - 1$.

- (e) Evaluer le critère d'arrêt².
- (f) Calculer $v_{k+1}(n)$, $n = 0, \dots, N - 1$ suivant (3.54).

Dans le cas linéaire une approche semblable a été considérée par [63]. Les auteurs sélectionnent les pôles en filtrant les dynamiques du système de manière itérative. Une fois la base construite les coefficients de Fourier sont ensuite déterminés. Il apparaît donc que dans son principe cette méthode est très différente de celle que nous présentons. L'approche analytique que nous avons présentée n'est donc pas l'extension de [63] au cas des modèles de Volterra.

²Comme dans la procédure de recherche exhaustive on peut utiliser AIC ou BIC.

3.5 Approche hybride de réduction de complexité d'un modèle de Volterra

Dans ce paragraphe, nous allons combiner les deux approches de réduction de complexité d'un modèle de Volterra. Nous allons montrer comment mettre à profit la décomposition matricielle et la décomposition sur base de fonctions orthogonales rationnelles.

Considérons un noyau de Volterra h_p causal, décomposé sur une BFOR. La meilleure approximation de ce noyau nécessite un nombre infini de fonctions de bases. Pour des raisons pratiques évidentes, ce développement est tronqué à un ordre fini. Cependant, quel est l'ordre de troncature optimal ? Dans la section précédente, nous avons utilisé un critère de sélection de modèle pour stopper la construction de notre base. Il s'avère que certains termes non significatifs demeurent d'où la nécessité d'une procédure de seuillage en aval. Il est aussi possible de faire un choix arbitraire de la BFOR tout en choisissant un ordre de troncature suffisamment élevé pour réduire l'erreur d'approximation du système. Une certaine redondance est introduite par ces termes supplémentaires. S'il est possible de grouper les coefficients de Fourier dans une matrice, nous pouvons alors utiliser les techniques de décomposition matricielle pour réduire davantage la complexité paramétrique.

Dans ce paragraphe, nous n'utiliserons pas la représentation triangulaire des coefficients de Fourier. Toutefois, à l'instar de la représentation matricielle des noyaux de Volterra que nous avons présentée, le modèle de Volterra-BFOR homogène, tronqué à l'ordre K_p peut aussi s'écrire :

$$y_p(n) = S_l^T(n) \mathbf{G}_{l,p-l} S_{p-l}(n) \quad (3.55)$$

où $S_p(n)$ est un vecteur de dimension K_p^p contenant tous les produits croisés d'ordre p des entrées filtrées $s_k(n)$ et $\mathbf{G}_{l,p-l}$ une matrice de dimension $K_p^l \times K_p^{p-l}$ telle que son (i, j) -ème élément soit le coefficient de Fourier qui pondère le produit de la i -ème composante de S_l et de la j -ème composante de S_{p-l} dans (3.55). Par exemple, dans le cas des noyaux quadratique et cubique nous avons respectivement :

$$\mathbf{G}_{1,1} = \begin{pmatrix} g_{0,0} & g_{0,1} & \cdots & g_{0,K_2-1} \\ g_{1,0} & g_{1,1} & \cdots & g_{1,K_2-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{K_2-1,0} & g_{K_2-1,1} & \cdots & g_{K_2-1,K_2-1} \end{pmatrix}, \quad S_1(n) = \begin{pmatrix} s_0(n) \\ s_1(n) \\ \vdots \\ s_{K_2}(n) \end{pmatrix}$$

$$\mathbf{G}_{2,1} = \begin{pmatrix} g_{0,0,0} & g_{0,0,1} & \cdots & g_{0,0,K_3-1} \\ g_{0,1,0} & g_{0,1,1} & \cdots & g_{0,1,K_3-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{K_3-1,K_3-1,0} & g_{K_3-1,K_3-1,1} & \cdots & g_{K_3-1,K_3-1,K_3-1} \end{pmatrix},$$

$$S_1(n) = \begin{pmatrix} s_0(n) \\ s_1(n) \\ \vdots \\ s_{K_2}(n) \end{pmatrix}, \quad S_2(n) = \begin{pmatrix} s_{0,0}(n) \\ s_{0,1}(n) \\ \vdots \\ s_{K_2,K_2}(n) \end{pmatrix}$$

Appliquons la décomposition SVD à la matrice $\mathbf{G}_{l,p-l}$. nous obtenons :

$$\mathbf{G}_{l,p-l} = \sum_{i=1}^r \sigma_i \mathbf{W}_i \mathbf{V}_i^T$$

Relation qui donne :

$$y_p(n) = \sum_{i=1}^r \sigma_i [S_l^T(n) \mathbf{W}_i] [\mathbf{V}_i^T S_{p-l}(n)] = \sum_{i=1}^r \sigma_i y_{l,i}(n) y_{p-l,i}(n)$$

L'intérêt de cette décomposition est davantage visible dans le cas d'un modèle de Volterra à noyau séparable. Dans ce cas, nous avons montré que la propriété de séparabilité se répercutait sur les coefficients de Fourier (voir l'équation (3.3)). En conséquence, la matrice $\mathbf{G}_{l,p-l}$ est donc de rang égal à un. D'où :

$$\mathbf{G}_{l,p-l} = \sigma \mathbf{W} \mathbf{V}^T$$

et par suite

$$y_p(n) = \sigma [S_l^T(n) \mathbf{W}] [\mathbf{V}^T S_{p-l}(n)]$$

Dans le cas particulier d'un modèle quadratique symétrique, $\mathbf{W} = \mathbf{V}$ d'où :

$$y_2(n) = \sigma [S_1^T(n) \mathbf{W}]^2$$

Ce modèle n'est caractérisé que par $(K_2 + 1)$ paramètres !

Algorithme de type NLMS pour l'estimation paramétrique d'un modèle de Volterra BFOR-SVD quadratique homogène : Considérons la fonction de coût

$$J(n) = e^2(n), \text{ avec } e(n) = d(n) - \sigma(n) [S_1^T(n) \mathbf{W}(n)]^2$$

La procédure de descente du gradient nous permet d'écrire, en posant $z(n) = S_1^T(n) \mathbf{W}(n)$:

$$\sigma(n+1) = \sigma(n) + \mu e(n) z^2(n) \quad (3.56)$$

$$\mathbf{W}(n+1) = \mathbf{W}(n) + 2\mu e(n) \sigma(n) z(n) S_1(n) \quad (3.57)$$

Le développement de Taylor de l'erreur e au premier ordre donne [83] :

$$e(n+1) = e(n) + \sum_{k=0}^{K_2-1} \frac{\partial e(n)}{\partial w_k(n)} \Delta w_k(n) + \frac{\partial e(n)}{\partial \sigma(n)} \Delta \sigma(n) \quad (3.58)$$

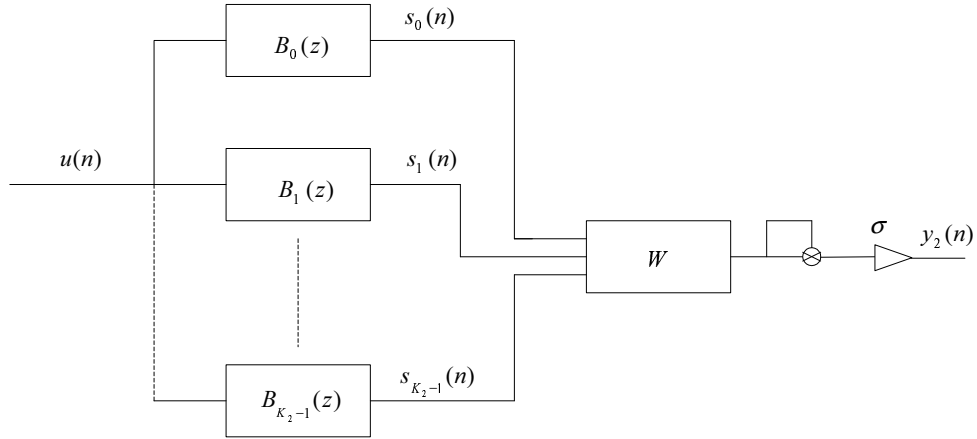


FIG. 3.3 – Représentation d'un modèle de Volterra BFOR-SVD quadratique homogène à noyau séparable

où : $\Delta w_k(n) = w_k(n+1) - w_k(n)$ et $\Delta \sigma(n) = \sigma(n+1) - \sigma(n)$. Sachant que :

$$\frac{\partial e(n)}{\partial w_k(n)} = -2\sigma(n)z(n)s_k(n) \quad \frac{\partial e(n)}{\partial \sigma(n)} = -z^2(n)$$

et tenant compte de (3.56) et de (3.57), on a alors

$$\begin{aligned} e(n+1) &= e(n) - 4\mu e(n)\sigma^2(n)z^2(n) \sum_{k=0}^{K_2-1} s_k^2(n) - \mu z^4(n)e(n) \\ &= e(n) \left[1 - \mu \left(4\sigma^2(n)z^2(n) \|S_1\|^2 + z^4(n) \right) \right] \end{aligned}$$

L'erreur $e(n+1)$ s'annule pour :

$$\mu_{opt} = \frac{1}{4\sigma^2(n)z^2(n) \|S_1\|^2 + z^4(n)}$$

Cependant, pour tenir compte des termes négligés dans le développement de Taylor, nous pouvons choisir :

$$\mu_{opt} = \frac{\mu}{4\sigma^2(n)z^2(n) \|S_1\|^2 + z^4(n)}, \quad 0 < \mu < 1 \quad (3.59)$$

D'où :

$$\sigma(n+1) = \sigma(n) + \mu \frac{e(n)z^2(n)}{4z^2(n)\sigma^2(n) \|S_1(n)\|^2 + z^4(n)} \quad (3.60)$$

$$W(n+1) = W(n) + 2\mu \frac{e(n)\sigma(n)z(n)S_1(n)}{4z^2(n)\sigma^2(n) \|S_1(n)\|^2 + z^4(n)} \quad (3.61)$$

Afin d'éviter des fluctuations trop importantes du facteur de normalisation nous allons plutôt considérer un filtrage passe-bas de celui-ci. Ce qui nous donne :

$$\sigma(n+1) = \sigma(n) + \mu \frac{e(n)z^2(n)}{v(n)} \quad (3.62)$$

$$W(n+1) = W(n) + 2\mu \frac{e(n)\sigma(n)z(n)S_1(n)}{v(n)} \quad (3.63)$$

$$v(n) = \beta v(n-1) + (1-\beta) \left(4z^2(n)\sigma^2(n) \|S_1(n)\|^2 + z^4(n) \right) \quad (3.64)$$

où β est une constante prise entre zéro et un.

Dans le cas d'un développement sur une base de Laguerre, nous pouvons combiner cet algorithme avec celui de l'estimation du pôle de Laguerre de manière adaptative. Nous avons alors :

Algorithme 3.8 : Approche hybride

1. Initialiser le pôle de Laguerre, $\sigma(0)$ et le vecteur $W(0)$;
2. Pour $n \geq 1$:
 - (a) Déterminer $\sigma(n)$ et le vecteur $W(n)$ à partir des équations (3.62-3.64) ;
 - (b) Calculer $G_2(n) = \sigma(n)W(n)W^T(n)$ (les composantes de $G_2(n)$ sont les coefficients de Fourier) ;
 - (c) Déterminer un nouveau pôle de Laguerre à partir des coefficients de Fourier calculés à l'étape précédente en utilisant l'algorithme décrit en 3.3.3.

3.6 Simulations

Dans cette section nous allons illustrer, à travers deux exemples, les performances des algorithmes que nous avons présentés. Nous évaluerons la procédure de sélection du pôle optimal de Laguerre ainsi que les approches exhaustives et analytiques de sélection des BFOR.

3.6.1 Cas linéaire

Nous considérons le modèle de fonction de transfert suivante [82] :

$$H(z) = \frac{3.8z^5 - 3.71z^4 + 0.3995z^2 - 0.2439z^2 + 0.254z}{z^5 - 0.85z^4 - 0.555z^3 + 0.616z^2 - 0.0733z - 0.0612}$$

dont la représentation des zéros et des pôles dans le plan complexe est donnée par la figure 3.4.

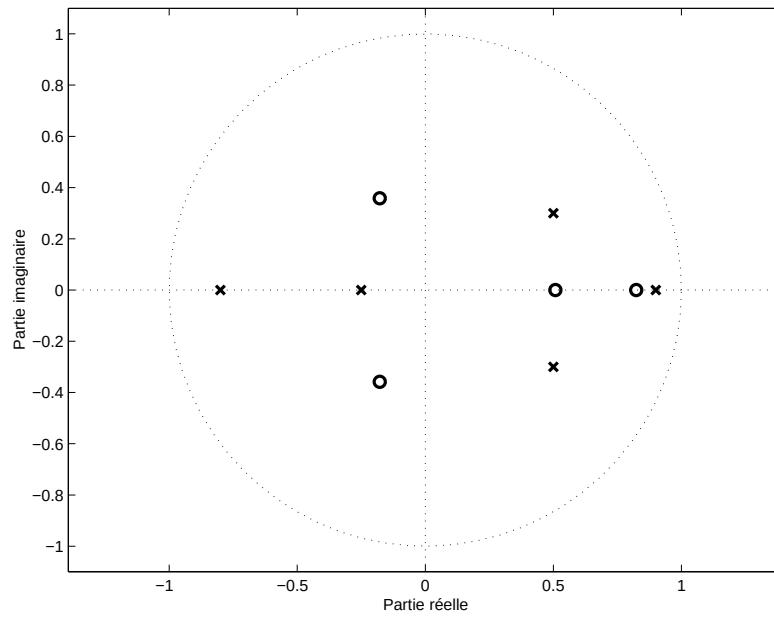


FIG. 3.4 – Lieu des pôles et des zéros du système simulé

A la sortie du système, le bruit additif est blanc et gaussien . En appliquant la procédure de construction que nous avons proposée au paragraphe 3.4.1, pour un nombre de données $N = 500$ et un rapport signal à bruit de $30dB$, avec 20 expériences indépendantes, l'ordre de troncature optimal au sens du critère BIC est : $K = 6$. Les pôles sélectionnés sont ceux donnés par le tableau 3.1.

Pôle	Valeur moyenne	Ecart type ($\pm$)
ξ_0	-0.740	$1.0 \cdot 10^{-5}$
ξ_1	0.619	$2.0 \cdot 10^{-3}$
ξ_2	-0.751	$5.0 \cdot 10^{-3}$
ξ_3	0.542	$6.0 \cdot 10^{-3}$
ξ_4	0.932	$4.0 \cdot 10^{-3}$
ξ_5	-0.089	$1.3 \cdot 10^{-2}$

TAB. 3.1 – Pôles sélectionnés

L'écart-type autour des valeurs moyennes des pôles sélectionnés croît avec la dimension de la base. La figure 3.5 montre l'évolution de l'erreur minimisée en fonction de l'ordre de troncature. Les premiers pôles choisis sont ceux qui contribuent le plus à la réduction de la dite erreur. Par conséquent, leurs coefficients de Fourier sont plus significatifs que ceux associés aux pôles suivants (figure 3.6).

Par ailleurs il est à noter que le gain statique correspondant au système reconstruit à l'aide des fonctions de base et des coefficients de Fourier estimés est égal à 7.102 ± 0.086 . Il est assez proche du gain statique du système réel qui est égal à 6.5307 .

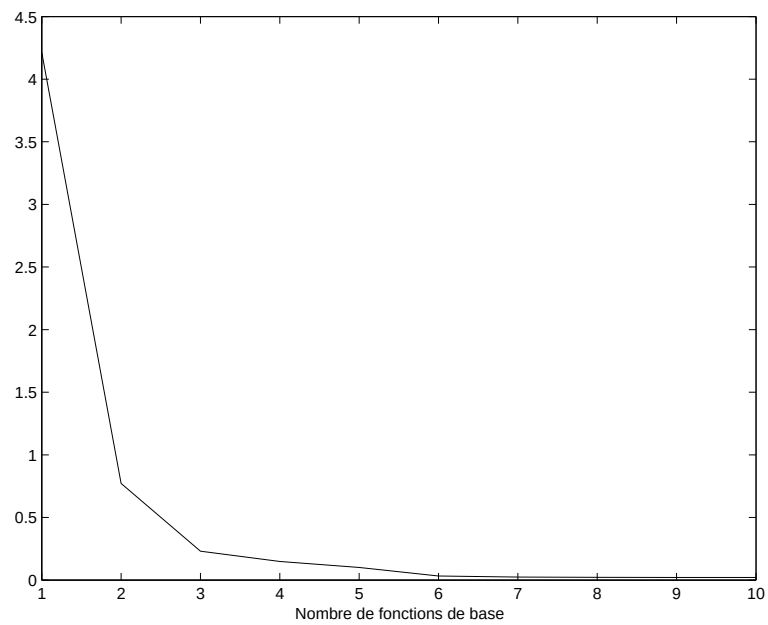


FIG. 3.5 – Erreur quadratique normalisée par le nombre de données en fonction de l'ordre de troncature

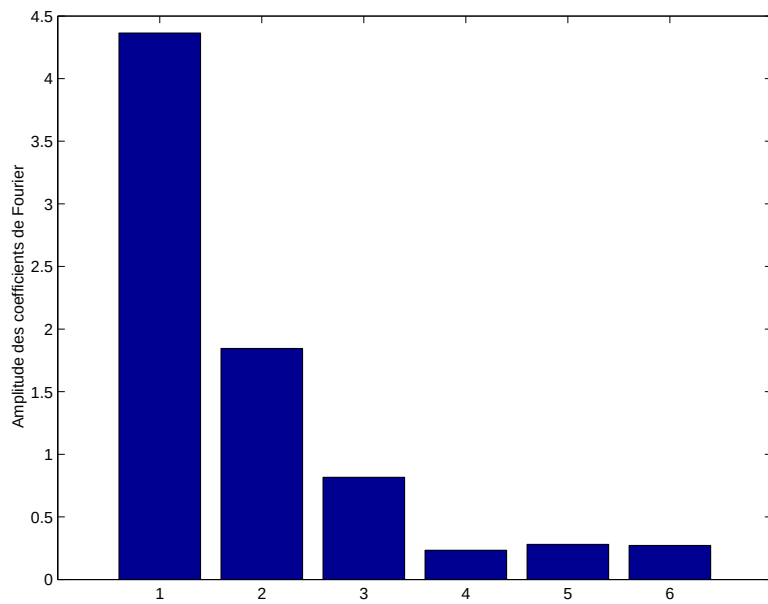


FIG. 3.6 – Coefficients de Fourier de la base sélectionnée

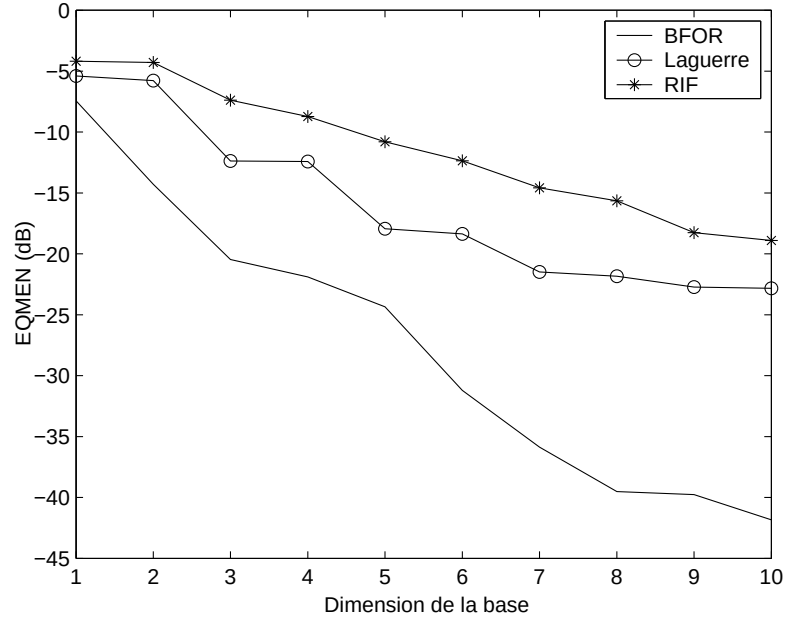


FIG. 3.7 – Comparaison des développements sur les bases RIF, Laguerre et BFOR

A présent, comparons le développement obtenu avec un développement sur une base RIF et sur une base de Laguerre dont le pôle est optimisé suivant l'algorithme de Fu et Dumont. Ce pôle optimal est $\xi_{opt} = -0.408$.

On peut remarquer, sur la figure 3.7, qu'à ordre de troncature identique, la BFOR approxime mieux le système simulé malgré le fait que seuls des pôles réels aient été utilisés. La précision de reconstruction est donnée par la relation suivante :

$$EQMEN = 10 \log \frac{\sum_{i=0}^{N-1} (h(i) - h_{est}(i))^2}{\sum_{i=0}^{N-1} h^2(i)}$$

où h est la réponse impulsionnelle du système et h_{est} celle reconstruite à partir du développement sur la BFOR.

Comme indiqué précédemment, l'ordre de sélection des pôles est important dans la procédure que nous avons développée. Cependant, quelle est la relation possible avec la sélection de la meilleure base au sens de l'entropie minimale telle que proposée par Bodin ? (voir le paragraphe 2.5)

A partir de la base sélectionnée nous avons appliqué la méthode de Bodin. Le tableau 3.2 donne la valeur de l'entropie pour la base que nous avons sélectionnée ainsi que pour la meilleure et la plus mauvaise base au sens de l'entropie minimale.

Les coefficients de Fourier correspondant à la meilleure base et à la pire base (au sens de l'entropie minimale) sont données par les figures 3.8 et 3.9

Base	Entropie	Ordre des pôles par rapport à la base initiale
Base sélectionnée	-12.976	(1,2,3,4,5,6)
Meilleure base	-13.451	(3,1,4,2,6,5)
Pire base	-8.292	(5,2,4,6,1,3)

TAB. 3.2 – Valeur de l’entropie pour différentes configurations de l’emplacement des pôles

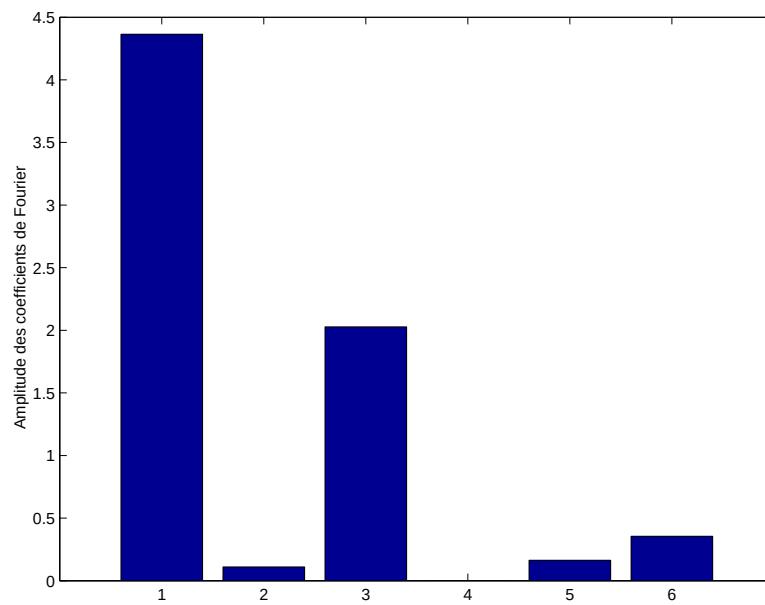


FIG. 3.8 – Coefficients de Fourier de la meilleure base au sens de l’entropie minimale

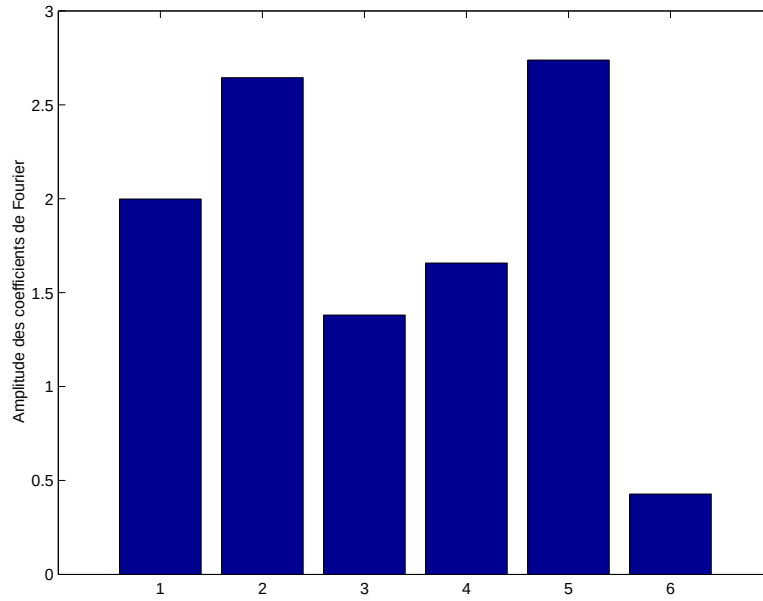


FIG. 3.9 – Coefficients de Fourier de la plus mauvaise base au sens de l'entropie minimale

Comme on peut le constater, au niveau de la meilleure base, l'énergie du système est concentrée sur quelques coefficients de Fourier. On peut donc aisément déterminer les coefficients les moins significatifs en vue d'un seuillage. En appliquant la règle de Donoho (2.62), seul le quatrième coefficient de la meilleure base est seuillé. Par ailleurs, il est à noter que la base sélectionnée par notre approche présente aussi la propriété d'avoir l'énergie du système concentrée sur quelques coefficients, les premiers. Son entropie est proche de celle de la meilleure base.

3.6.2 Cas non-linéaire

Nous considérons deux systèmes qui ne diffèrent que par la présence ou non d'au moins un retard unitaire. Ils sont décrits par des modèles de Volterra du second ordre :

– Système $\mathcal{S}_1$:

noyau du premier ordre

$$H_1(z) = \frac{z(z+0.5)}{(z-0.3)(z-0.2)} \quad (3.65)$$

noyau du second ordre

$$h_2(i, j) = 0.25h(i)h(j) \quad \text{avec} \quad h(i) = \mathcal{Z}^{-1} \left\{ \frac{z(z+1)}{(z-0.8)(z+0.8)} \right\} \quad (3.66)$$

– Système $\mathcal{S}_2$:

noyau du premier ordre

$$H_1(z) = \frac{z + 0.5}{(z - 0.3)(z - 0.2)} \quad (3.67)$$

noyau du second ordre

$$h_2(i, j) = 0.25h(i)h(j) \quad \text{avec} \quad h(i) = \mathcal{Z}^{-1} \left\{ \frac{z + 1}{(z - 0.8)(z + 0.8)} \right\} \quad (3.68)$$

La mémoire du modèle est choisie égale à $M = 20$, ce qui correspond à 420 coefficients. Si l'on tient compte de la symétrie du noyau quadratique, la représentation triangulaire du modèle contient 230 paramètres. L'entrée des deux systèmes est un signal blanc, gaussien, centré et de variance unité. La sortie respective des deux systèmes est perturbée par un bruit additif blanc, gaussien, indépendant du signal d'entrée.

3.6.2.1 Choix du pôle optimal de Laguerre

Dans ce paragraphe, nous évaluons les performances des algorithmes d'optimisation du pôle de Laguerre présentés dans la section 3.3. Le système $\mathcal{S}_1$ est développé sur des bases de Laguerre. Rappelons que les algorithmes que nous avons présentés permettent l'estimation paramétrique du modèle de Volterra-Laguerre tout en optimisant les bases utilisées. L'optimisation des bases est, ici, vue au sens du choix du pôle optimal.

Nous utilisons $N = 5000$ données d'entrée-sortie. Le rapport signal à bruit est égal à 30dB. Le système $\mathcal{S}_1$ ne répond pas aux conditions requises par l'algorithme de [21]. En appliquant le dit algorithme à $\mathcal{S}_2$, les pôles optimaux sont respectivement : $\xi_{1,opt} = 0.525$ et $\xi_{2,opt} = 0.734$. Sachant que $\mathcal{S}_1$ et $\mathcal{S}_2$ ont les mêmes propriétés dynamiques, nous pouvons comparer ces valeurs à celles fournies par les méthodes que nous avons développées. Par ailleurs, nous pouvons aussi tracer les courbes théoriques des fonctions J_1 et J_2 (voir la figure 3.10). Ces fonctions ont chacune un minimum unique en $\xi_{1,opt} = 0.525$ et $\xi_{2,opt} = 0.733$ respectivement.

Estimation en bloc : Nous étudions le comportement de l'algorithme 3.1 suivant trois aspects : l'ordre de troncature K du développement des noyaux des Volterra sur les bases de Laguerre, l'initialisation des pôles de Laguerre et le critère d'arrêt de l'algorithme.

Afin d'étudier l'effet de l'ordre de troncature K , choisi *a priori*, sur l'optimisation des pôles ξ_1 (partie linéaire) et ξ_2 (partie quadratique), nous initialisons les pôles de Laguerre par $\xi_p = 0.001$, $p = 1, 2$. Les résultats obtenus sont reportés sur le tableau 3.3. La figure 3.11 illustre le comportement des pôles estimés durant la procédure proposée.

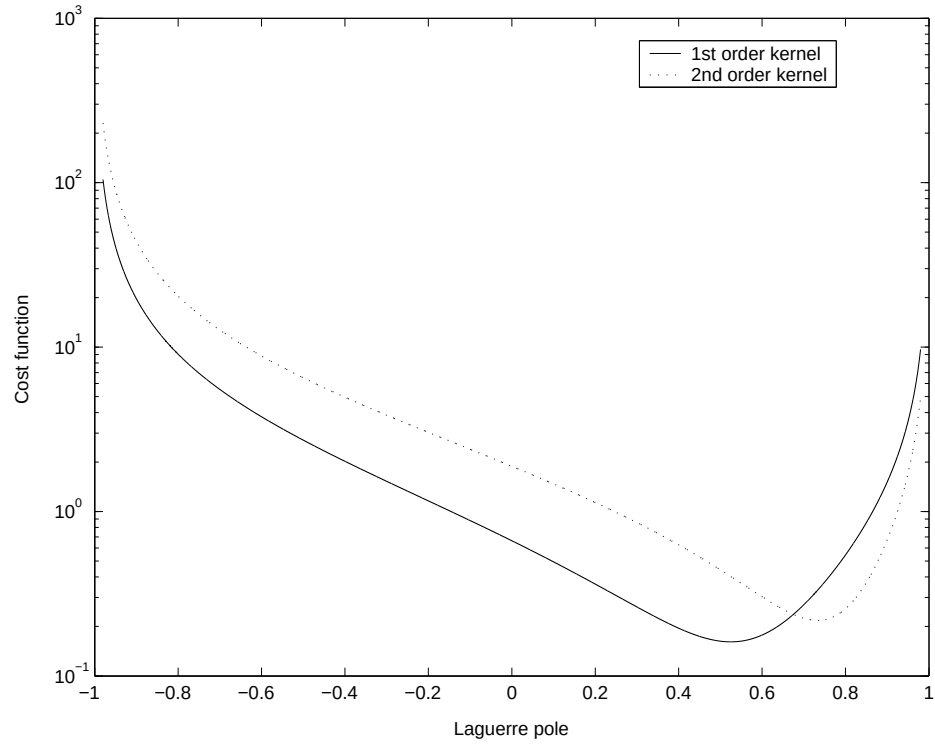


FIG. 3.10 – Courbes théoriques des fonctions de coût J_1 (trait plein) et J_2 (trait discontinu)

K	ξ_1	ξ_2	$EQMN(dB)$
3	0.543 ± 0.013	0.820 ± 0.012	-9.394 ± 0.443
4	0.573 ± 0.018	0.824 ± 0.014	-12.010 ± 0.587
5	0.540 ± 0.013	0.806 ± 0.006	-14.495 ± 0.199
6	0.526 ± 0.011	0.819 ± 0.003	-14.868 ± 0.182
7	0.511 ± 0.011	0.812 ± 0.006	-15.212 ± 0.220
8	0.517 ± 0.010	0.815 ± 0.003	-15.377 ± 0.219
9	0.504 ± 0.010	0.813 ± 0.005	-15.645 ± 0.195
10	0.509 ± 0.008	0.801 ± 0.003	-15.870 ± 0.165

TAB. 3.3 – Valeurs optimales des pôles en fonction de l'ordre de troncature

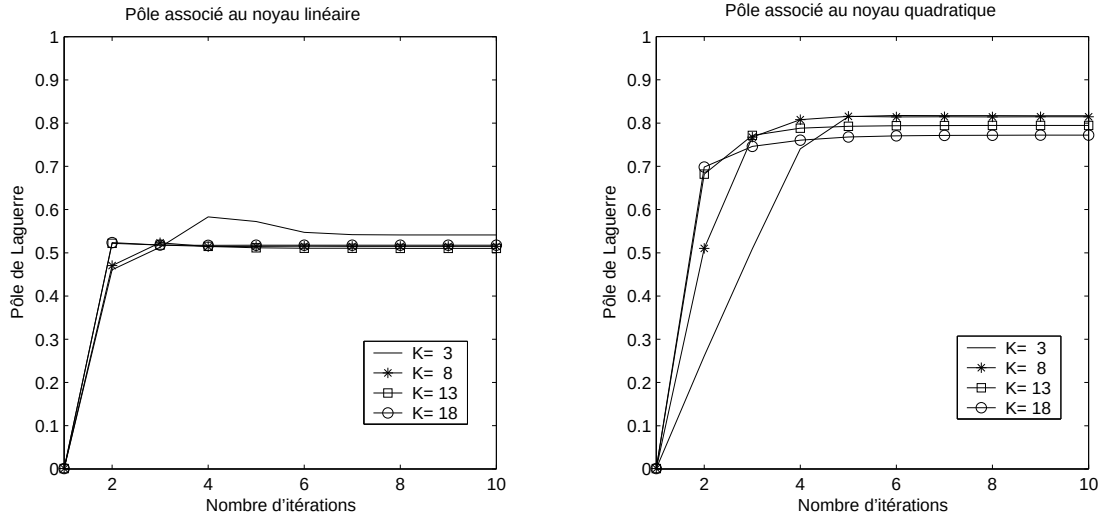


FIG. 3.11 – Estimation des pôles

Sur la figure 3.11, nous pouvons noter que l'algorithme converge en quelques itérations. Pour un ordre de troncature significatif ($K = 18$ par exemple), une itération suffit pour l'obtention de la valeur du pôle optimal. Pour des ordres de troncature plus faibles, il est nécessaire de recourir à davantage d'itérations. Sur le tableau 3.3, nous observons que les valeurs des pôles varient très peu avec l'ordre de troncature. Si donc l'objectif de l'étude est simplement la détermination du pôle optimal, le choix arbitraire de l'ordre de troncature n'a pas d'effet significatif sur la valeur du pôle optimal obtenu. Toutefois, augmenter l'ordre de troncature permet d'augmenter la vitesse de convergence de l'algorithme mais avec un surcoût de calcul. Par contre, dans l'objectif de l'identification du système, nous pouvons constater, sur le même tableau 3.3, que l'ordre de troncature influe sur la précision du modèle. Nous reviendrons sur cet aspect du problème dans les simulations suivantes. Toutefois, nous pouvons noter que le fait d'avoir une procédure d'estimation itérative permet d'améliorer non seulement l'estimation des pôles de Laguerre mais aussi la précision globale du modèle estimé (voir la figure 3.12).

Voyons à présent (figure 3.13) l'effet de l'initialisation des pôles de Laguerre sur la vitesse de convergence de l'algorithme d'optimisation des pôles de Laguerre, pour un ordre de troncature fixé à $K = 7$.

Bien que les valeurs initiales des pôles soient différentes, l'algorithme converge vers la même valeur. Cependant, la convergence est d'autant plus rapide que le pôle initial est proche de la valeur optimale. une fois de plus nous pouvons noter que les pôles convergent vers leurs valeurs optimales en relativement peu d'itérations.

Par ailleurs, dans le développement théorique de l'algorithme, nous avons montré que lorsque la valeur du pôle atteint son optimum, $R_{2,p}$ s'annule (voir le paragraphe 3.3.2). La figure 3.14 montre l'évolution de $R_{2,1}$ et $R_{2,2}$. Nous notons que lorsque ces quantités deviennent quasiment nulles, les pôles ont atteint leurs valeurs optimales ; ce qui est en

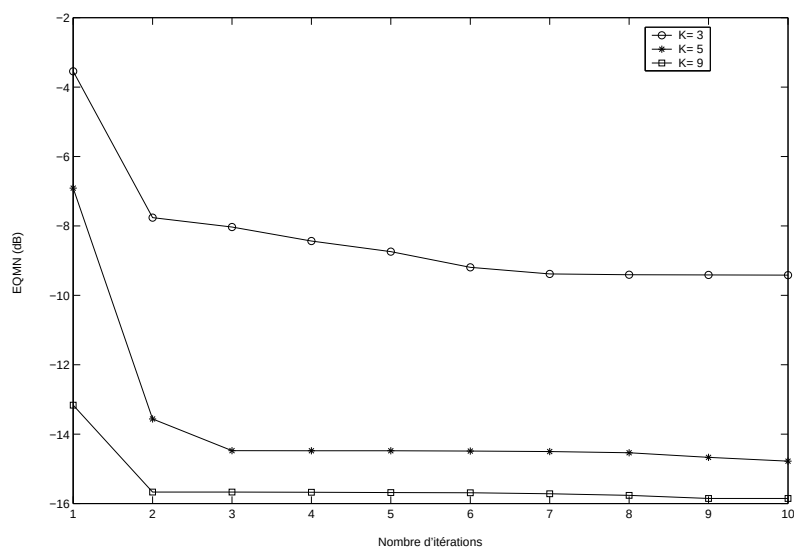


FIG. 3.12 – Evolution de l'EQMN pour différents ordres de troncature

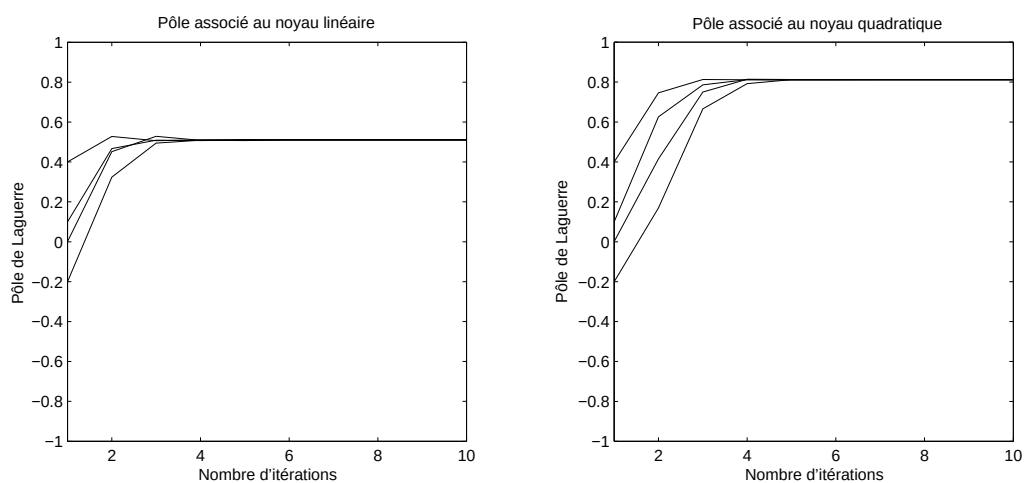


FIG. 3.13 – Estimation des pôles pour différentes valeurs initiales des pôles de Laguerre

parfaite conformité avec le résultat théorique.

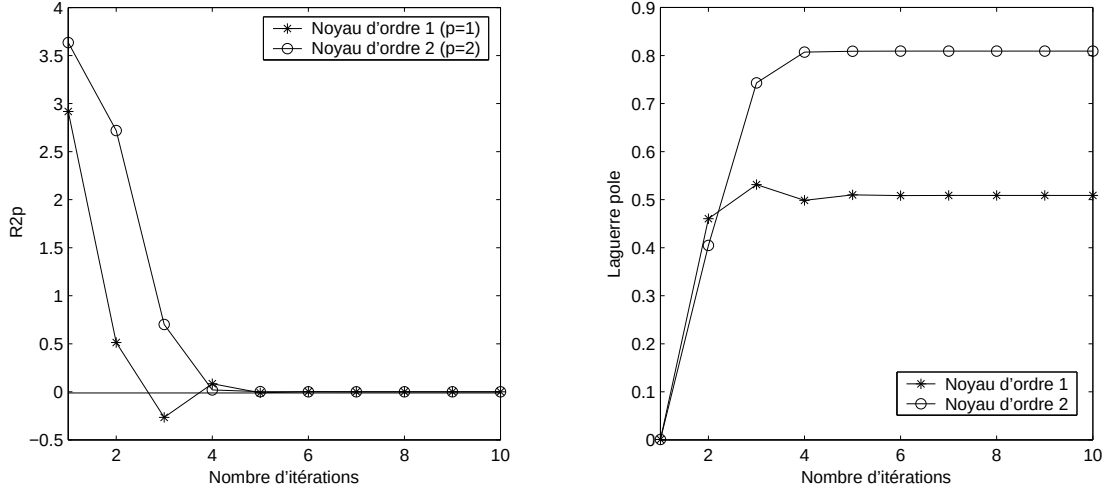


FIG. 3.14 – Estimation des pôles (à droite) et évolution de $R_{2,p}$ (à gauche)

De cette étude nous retiendrons les observations suivantes :

- Le choix arbitraire de l'ordre de troncature du développement sur une base de Laguerre influe sur la vitesse de convergence de l'algorithme et non pas sur la valeur optimale du pôle estimé.
- L'initialisation arbitraire du pôle de Laguerre influe sur la vitesse de convergence de l'algorithme et non sur la valeur optimale estimée.
- Très peu d'itérations suffisent pour obtenir les pôles optimaux.
- La quantité $R_{2,p}$ constitue une bonne mesure de la convergence de l'algorithme. Lorsqu'elle devient quasiment nulle, le pôle ξ_p a convergé vers sa valeur optimale.
- Les pôles optimaux obtenus sont proches de ceux donnés par l'algorithme de [21] pour le système $\mathcal{S}_2$.

Estimation adaptative : Nous allons à présent évaluer l'algorithme adaptatif 3.2 que nous avons proposé dans le paragraphe 3.3.3. Les conditions expérimentales sont les mêmes que dans le cas de l'estimation en bloc. Les valeurs initiales des pôles de Laguerre sont prises égales à zéro. Le développement des noyaux sur les bases de Laguerre est tronqué à $K = 7$; en conséquence le filtre de Volterra-Laguerre résultant a seulement 35 coefficients de Fourier à estimer alors que le filtre de Volterra standard en a 230. Le pas d'adaptation de l'algorithme NLMS est pris égal 1. Les seuils de convergence sont choisis égaux à $\varepsilon_p = 10^{-3}$, $p = 1, 2$ tandis que la largeur de la fenêtre d'observation est $N_O = 50$.

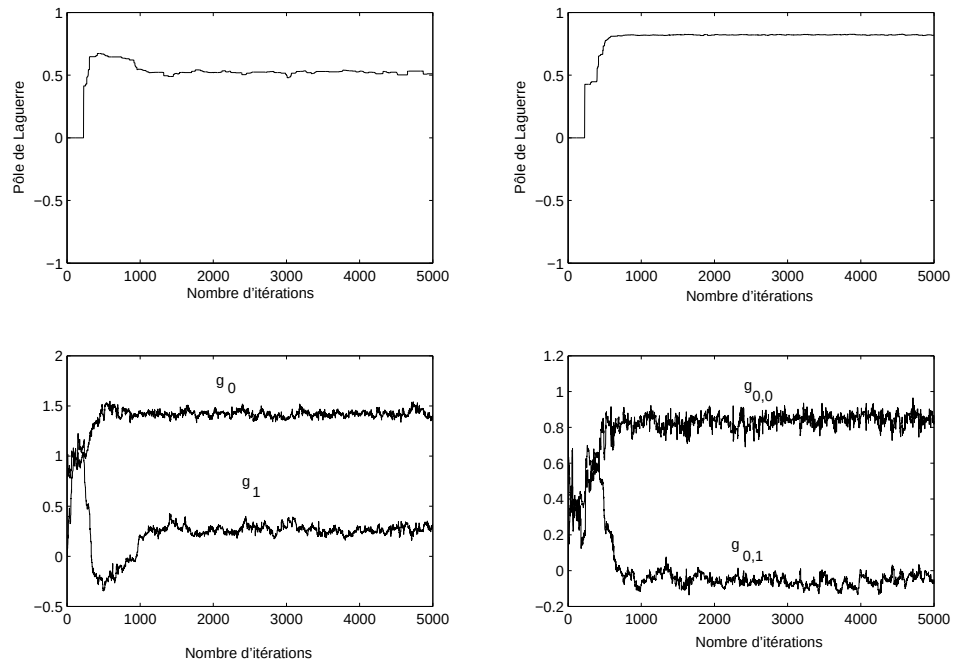


FIG. 3.15 – Estimation des pôles de Laguerre (en haut) et de quelques coefficients de Fourier coefficients (en bas) associés au noyau du premier ordre (à gauche) et du second ordre (à droite)

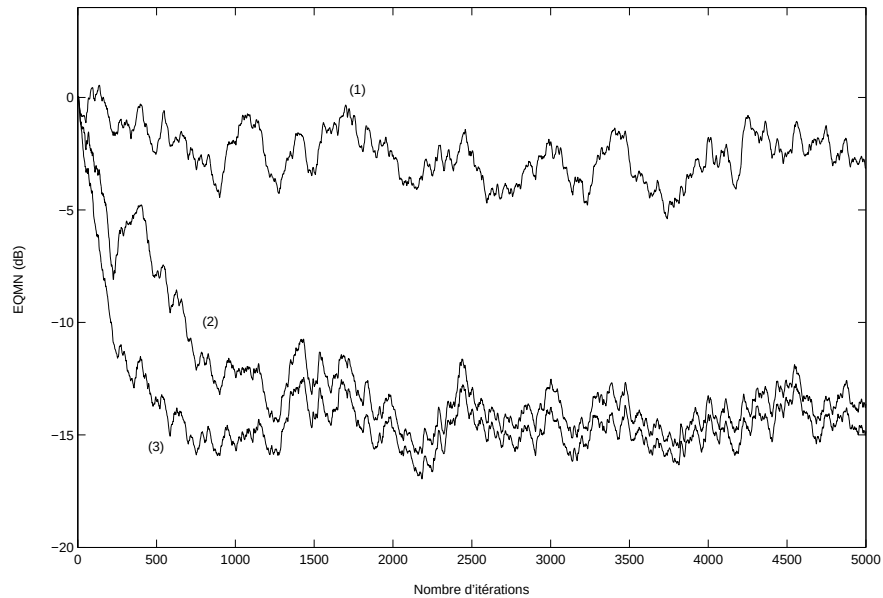


FIG. 3.16 – Comparaison de trois configurations du filtre de Volterra-Laguerre : (1) Pôles de Laguerre arbitrairement fixés (2) Pôles de Laguerre estimés avec l'algorithme proposé (3) Pôles de Laguerre égaux aux pôles optimaux

L'évolution des pôles de Laguerre estimés ainsi que celle de quelques coefficients de Fourier est illustrée par la figure 3.15. Nous constatons que l'algorithme converge bel et bien, et ce vers les valeurs des pôles obtenus dans le cas de l'estimation en bloc. Une question importante est, cependant, celle du choix du seuil de convergence ε_p et de la largeur N_0 de la fenêtre d'observation. Lorsque ε_p est choisi trop petit, un grand nombre d'itérations peut être nécessaire avant d'effectuer l'estimation du pôle optimal. Par conséquent, relativement au problème traité, une attention particulière devrait être portée sur le choix de ces paramètres.

Pour illustrer la convergence globale de la procédure d'identification, le tracé de l'EQMN est donné par la figure 3.16. Il est comparé aux tracés obtenus dans deux configurations différentes. La première a consisté à effectuer un choix arbitraire du pôle de Laguerre tandis que la seconde considère un choix optimal des pôles. Rappelons que les pôles optimaux sont ceux obtenus à l'aide du tracé de la courbe théorique. Nous constatons qu'en régime permanent, les performances de l'algorithme proposé sont similaires à celles obtenues avec les pôles optimaux obtenus grâce aux courbes théoriques. Evidemment l'algorithme proposé converge plus lentement.

Les pôles estimés tant dans la méthode adaptative que dans la méthode en bloc sont proches des pôles optimaux théoriques et de ceux obtenus par l'algorithme de Campello *et al*[21].

3.6.2.2 Approche exhaustive pour la sélection des BFOR

Considérons à présent le système $\mathcal{S}_2$. Le système est développé sur des BFOR. Nous utilisons un certain nombre de données pour la sélection des pôles. Puis, l'ensemble des données $N = 5000$ est utilisé pour évaluer les performances en terme d'EQMN. Nous étudions les questions de l'ordre de troncature optimal et de l'utilisation d'une base unique ou de deux bases distinctes. La procédure de seuillage est aussi évaluée.

Détermination de l'ordre de troncature : Nous comparons tout d'abord les ordres de troncature obtenus en utilisant comme critère d'arrêt de l'algorithme, AIC ou BIC lorsque les noyaux sont développés sur une base unique ou sur des bases différentes. Les tableaux 3.4 et 3.5 résument les résultats pour trois niveaux de bruit différents et pour différents nombres de données.

SNR	10 dB				20 dB				30 dB			
N	200	300	400	500	200	300	400	500	200	300	400	500
BIC	3	3	3	3	5	5	5	6	7	7	8	9
AIC	3	5	5	6	6	7	7	8	8	9	9	10

TAB. 3.4 – Ordre de troncature optimal d'une base unique

SNR	10 dB				20 dB				30 dB			
N	200	300	400	500	200	300	400	500	200	300	400	500
BIC	2	2	2	4	4	4	4	4	6	6	6	6
AIC	2	2	4	4	4	4	6	6	6	6	6	8

TAB. 3.5 – Ordre de troncature optimal des deux bases

Nous pouvons faire les observations suivantes :

- L'ordre de troncature augmente avec le rapport signal à bruit et avec le nombre de données ;
- Le critère AIC a tendance à surdimensionner les bases.

Tenant compte de ces observations, on peut noter qu'une attention particulière devrait être portée sur le nombre de données utilisées pour la sélection des bases. Il y a un risque de surdimensionnement de la base lorsque le nombre de données traitées augmente.

Dans la suite nous utiliserons le critère BIC comme critère d'arrêt de notre algorithme.

La figure 3.17 montre que l'ordre de troncature donné par le critère d'arrêt, est très proche de la valeur de K à partir de laquelle la variation de l' $EQMN$ (3.49) avec la dimension de la base n'est plus significative.

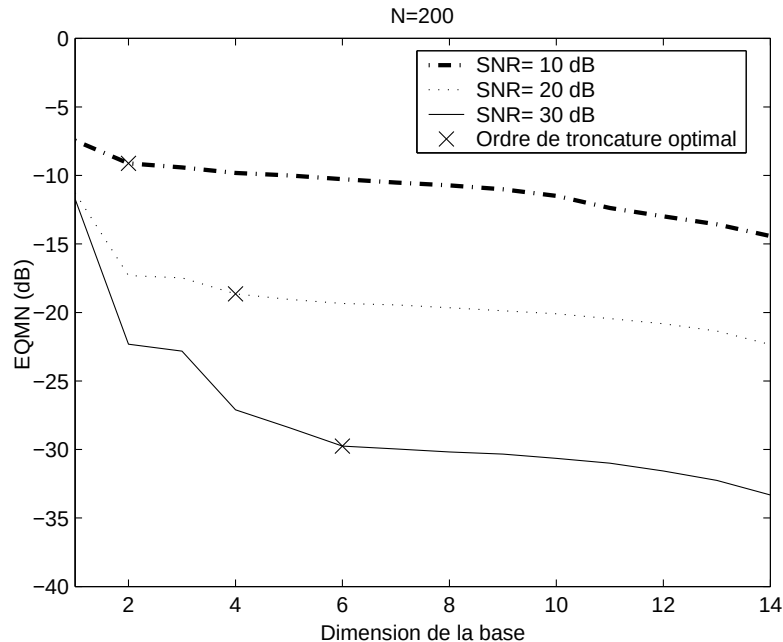


FIG. 3.17 – EQMN en fonction de la dimension de la base pour un développement sur deux bases

La sortie reconstituée à l'aide de la base optimale obtenue diffère très peu du signal réel comme en témoigne la figure 3.18.

3.6. Simulations

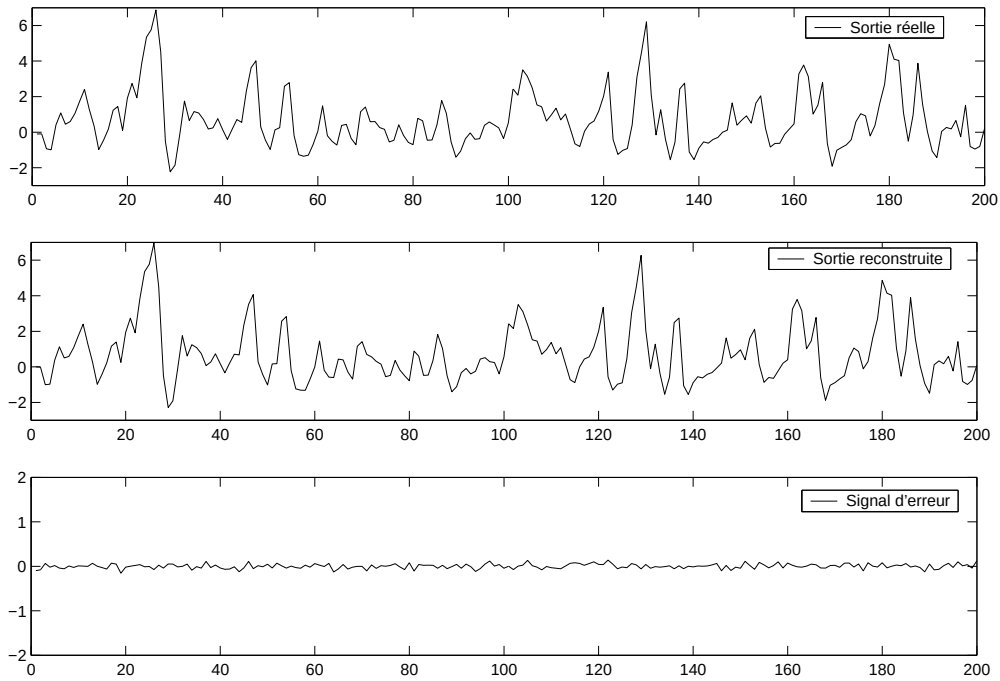


FIG. 3.18 – Comparaison de la sortie du système avec sa reconstruction obtenue à l'aide du modèle de Volterra-BFOR optimal (SNR=30 dB)

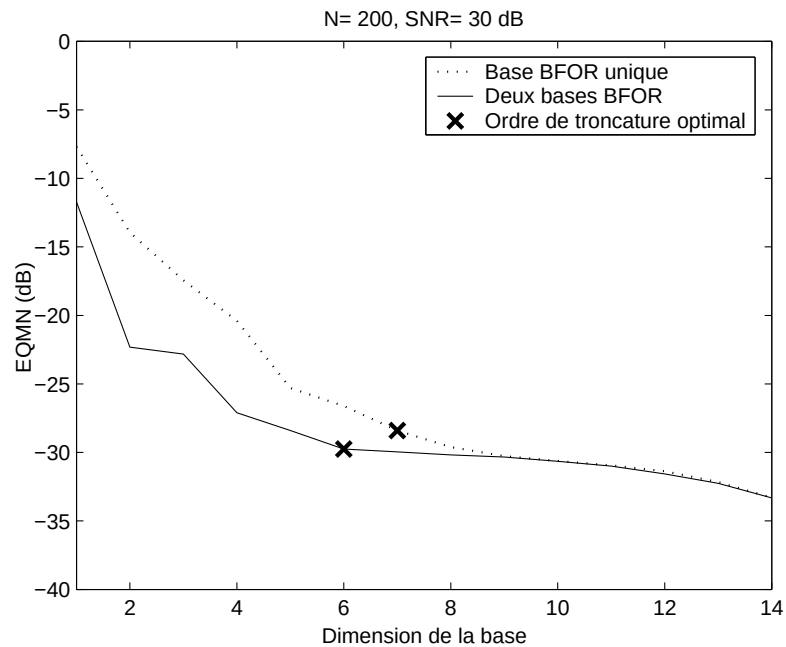


FIG. 3.19 – Comparaison des développements sur une base unique et sur des bases distinctes

Comparaison des développements sur une base unique et sur des bases distinctes :

En comparant les résultats des tableaux 3.4 et 3.5, on constate que lorsqu'une base unique est utilisée, l'ordre de troncature donné par les critères BIC et AIC est plus grand que dans le cas de l'utilisation de deux bases distinctes. Comme nous pouvons le voir au travers de la figure 3.19, pour un ordre de troncature faible la précision de la reconstruction du modèle, donnée par l'EQMN, est meilleure lorsque deux bases distinctes sont utilisées. Plus l'ordre de troncature augmente, plus les performances des deux représentations se rejoignent.

Comparaison avec un développement sur des bases de Laguerre : A présent, nous allons comparer les performances des représentations sur deux BFOR et sur des bases de Laguerre. Les pôles de Laguerre optimaux sont obtenus en appliquant la méthode de [21]. Dans le cas de l'utilisation d'une base unique, le pôle optimal de Laguerre est $\xi_{opt} = 0.59$. Pour deux bases distinctes, nous avons $\xi_{1,opt} = 0.525$ pour la partie linéaire et $\xi_{2,opt} = 0.734$ pour la partie quadratique.

Rappelons que pour $N = 200$ données utilisées lors de la sélection des BFOR et pour un niveau de bruit équivalent à $SNR = 30$ dB, l'ordre de troncature optimal est $K = 6$. La valeur correspondante de l'EQMN est égale à -29.7 dB. La figure 3.20 confirme

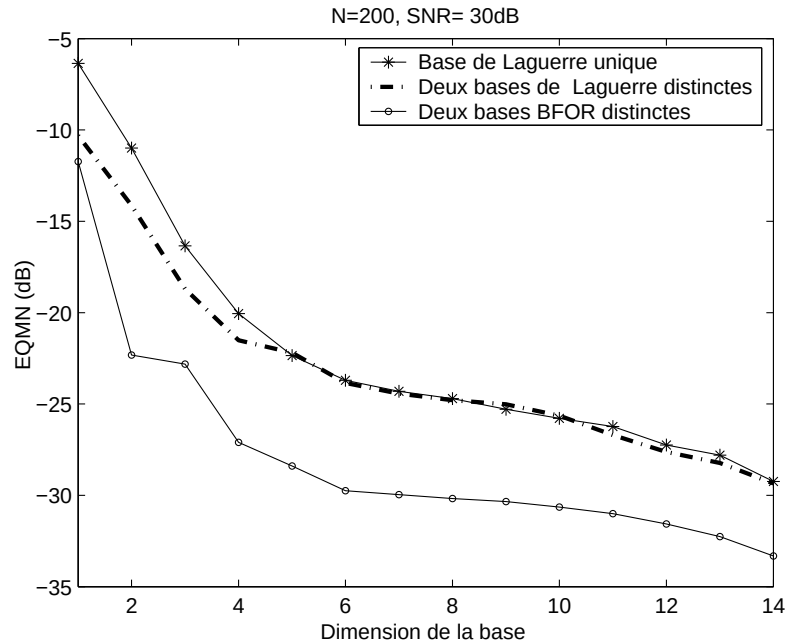


FIG. 3.20 – Comparaison des représentations sur des bases de Laguerre et des bases BFOR

l'amélioration des performances lorsque des bases BFOR sont utilisées en lieu et place des bases de Laguerre. En effet, on peut observer que pour atteindre la performance obtenue avec deux bases BFOR tronquées à $K = 6$, il faut utiliser des bases optimales de Laguerre tronquées à $K = 14$. Le tableau 3.6 montre la réduction de complexité obtenue grâce à un développement sur des BFOR.

Modèles de Volterra	Standard	Volterra-Laguerre	Volterra-BFOR
Nombre de paramètres	230	119	27

TAB. 3.6 – Complexité paramétrique

Technique de seuillage : Sur les figures 3.21 et 3.22, nous traçons l'EQMN pour différents ordres de troncature en distinguant les termes linéaire et quadratique. Pour les cinq valeurs de K_2 considérées (figure 3.21), l'EQMN devient relativement constante pour des valeurs de K_1 plus grandes ou égales à 6. Nous faisons la même observation sur la figure 3.22. En effet, pour les valeurs considérées de K_1 , l'EQMN ne décroît plus significativement pour $K_2 \geq 2$. Rappelons que l'ordre de troncature optimal donné par le critère BIC est $K = K_1 = K_2 = 6$, valeur appartenant à l'ensemble des couples (K_1, K_2) pour lesquels la variation de l'EQMN est peu significative. Notons aussi que pour $K_1 = 6$ si nous désirons une EQMN comprise entre -30 dB et -25 dB, nous pouvons prendre K_2 dans l'intervalle $[2; 8]$ (figure 3.22). Pour ces valeurs de K_1 et de K_2 , le nombre de paramètres est compris entre 11 et 42. En conséquence, nous pouvons prendre les plus petites valeurs de K_1 et de K_2 permettant d'atteindre l'EQMN désirée. C'est l'objet de la technique de seuillage (algorithme 3.6) que nous avons proposée.

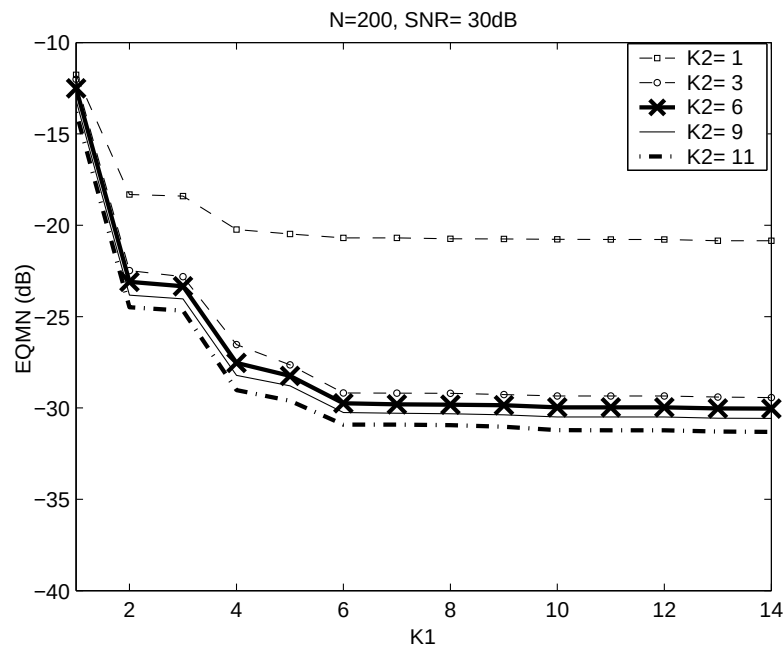


FIG. 3.21 – EQMN en fonction de l'ordre de troncature du développement du noyau du premier ordre

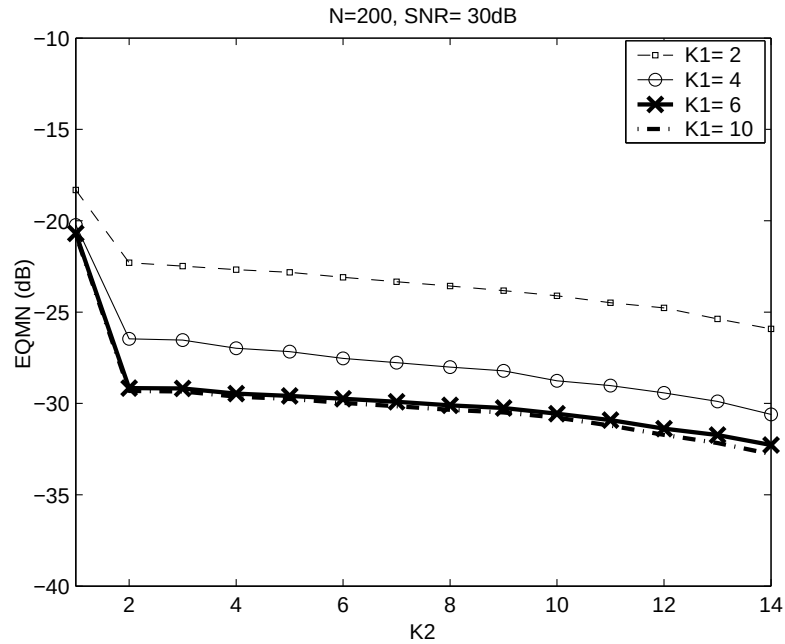


FIG. 3.22 – EQMN en fonction de l'ordre de troncature du développement du noyau du second ordre

Le tableau 3.7 montre que la réduction de complexité peut être réalisée sans trop détériorer le modèle. Par exemple, le nombre de paramètres est réduit d'approximativement 70% lorsque la technique de seuillage est appliquée avec $\kappa = 10$ alors que l'EQMN ne se détériore que d'environ 2.1 dB (table 3.7).

Seuillage	Avant	Après		
κ	-	70	30	10
EQMN(dB)	-29.7	-29.5	-29.2	-27.6
$C^{BFOR}(K_1, K_2)$	27	16	9	8
K_1	6	6	6	5
K_2	6	4	2	2

TAB. 3.7 – Technique de seuillage ($N = 200$ et $SNR = 30\text{dB}$)

3.6.2.3 Approche analytique

Dans ce paragraphe, nous reprenons le système décrit par les équations (3.65) et (3.66). Nous lui appliquons l'approche analytique pour la sélection des bases BFOR décrite au paragraphe 3.4.3 par l'algorithme 3.7. Nous utilisons les mêmes conditions expérimentales que dans le paragraphe 3.6.2.1. La figure 3.23 montre l'évolution de la puissance du signal résiduel, puissance qui correspond à l'erreur quadratique moyenne en sortie. Nous pouvons constater que celle-ci se stabilise autour de $K = 7$. Les processus de construction de la base et d'identification du système peuvent donc être arrêtés à ce niveau.

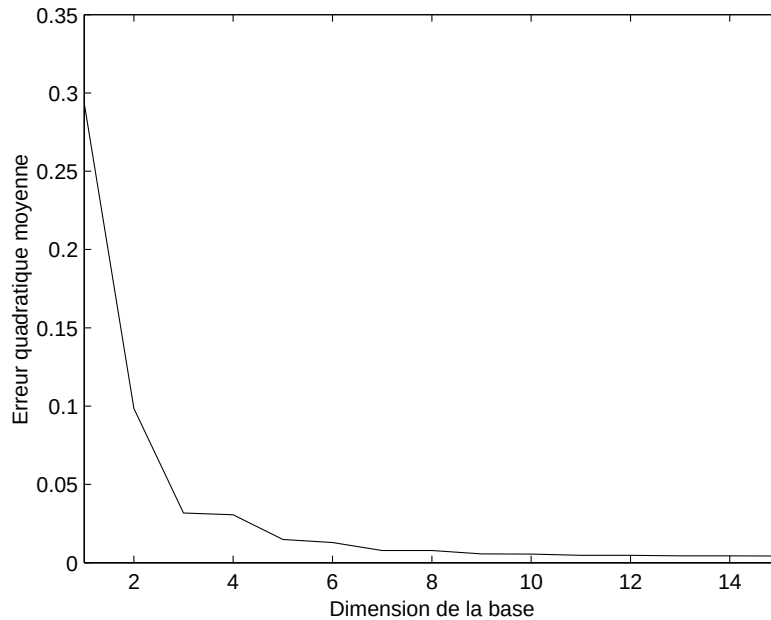


FIG. 3.23 – Puissance du signal résiduel en fonction de la dimension de la base

Selon la figure 3.24, l'approche exhaustive, comparée à l'approche analytique, donne de meilleures performances en terme d'EQMN. Rappelons que dans l'approche exhaustive la construction de la base se fait de sorte que l'erreur quadratique soit minimisée à chaque itération, ce qui n'est pas le cas pour l'approche analytique. Toutefois, l'approche analytique étant moins coûteuse en temps de calcul, elle constitue une bonne alternative à l'approche exhaustive.

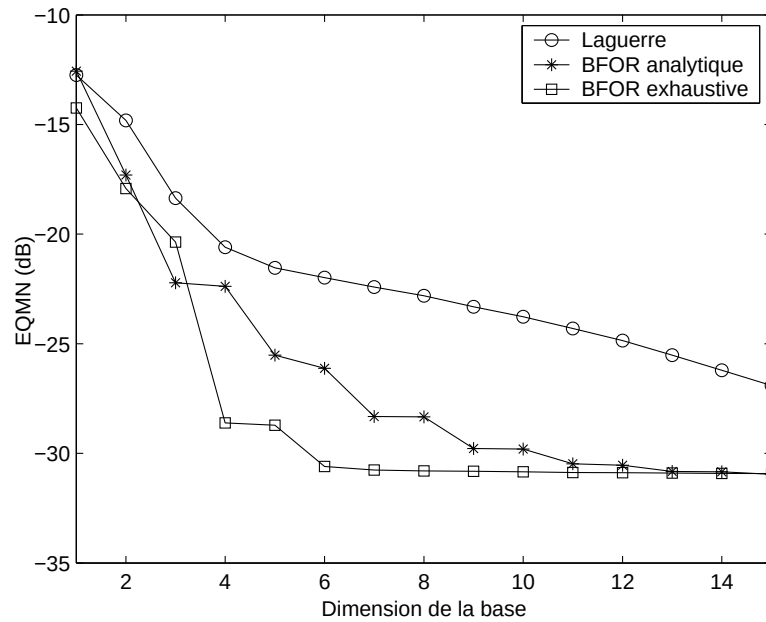


FIG. 3.24 – Comparaison des différentes approches

3.7 Conclusion

Dans ce chapitre nous avons montré que la complexité paramétrique d'un modèle de Volterra peut être réduite en utilisant un développement des noyaux sur des bases de fonctions orthogonales. Nous avons particulièrement étudié les représentations sur des bases de Laguerre et sur les bases beaucoup plus générales de type BFOR.

La base de Laguerre est caractérisée par un seul pôle. Son optimisation, au sens du choix du pôle, peut se faire de manière analytique. Campello *et al.* ont été les premiers à proposer une solution analytique à ce problème. En relaxant certaines des conditions qu'ils imposent au système qu'ils considèrent, nous avons proposé une nouvelle solution analytique pour une classe de modèles de Volterra plus large. Deux algorithmes basés sur la seule estimation des coefficients de Fourier ont été présentés. Le premier traite les entrées et sorties du modèle en bloc tandis que le second adapte à la fois les pôles (cas des bases distinctes pour chaque noyau) et les coefficients de Fourier. Ces algorithmes donnent de bonnes performances. L'ordre de troncature des bases est fixé *a priori*. Toutefois, les valeurs des pôles déterminés ne sont pas fonctions de l'ordre de troncature.

La sélection d'une base BFOR est beaucoup plus complexe. Dans ce chapitre, nous avons proposé une nouvelle procédure de sélection des pôles d'une BFOR. L'avantage de cette approche est que nous pouvons tenir compte de la connaissance *a priori* que nous avons sur le système. La recherche des pôles étant exhaustive, nous ne sommes pas soumis au problème des minima locaux. Toutefois, la méthode peut s'avérer très lourde. Nous avons alors présenté une alternative qui consiste à remplacer la recherche

3.7. Conclusion

exhaustive par la recherche du pôle optimal de Laguerre pour différents sous-systèmes. Nous avons appelé cette approche, approche analytique.

Enfin, une troisième approche a été proposée. Elle combine les approches de décomposition matricielle et la représentation sur les bases orthogonales. Dans le cas d'un noyau quadratique séparable, nous avons montré comment une réduction de complexité drastique pouvait être réalisée et avons présenté un algorithme adaptatif.

Chapitre 4

Méthodes orthogonales d'identification

4.1 Introduction

Dans ce chapitre nous présentons deux types de méthodes orthogonales d'identification. Le premier résulte de l'exploitation du formalisme des bases de polynômes orthogonaux tandis que le second s'appuie sur une orthogonalisation de Gram-Schmidt des entrées filtrées par les fonctions issues des BFOR. Un couplage des deux approches est aussi proposé dans le cas particulier d'une entrée gaussienne.

Dans le chapitre 1, il a été établi que l'estimation paramétrique d'un modèle de Volterra quadratique peut être complètement découplée, lorsque l'entrée considérée est indépendante et distribuée de manière identique (i.i.d., en anglais) et suit une distribution de probabilité symétrique. C'est le cas d'une entrée gaussienne ou encore de la plupart des signaux utilisés en télécommunications (PAM, QAM, PSK). Cependant, cette propriété ne se vérifie plus lorsqu'on augmente l'ordre de non-linéarité du modèle. En général, l'estimation des coefficients des noyaux de Volterra au sens de l'erreur quadratique moyenne minimale requiert le calcul de l'inverse de la matrice des moments de l'entrée et l'estimation de ces moments jusqu'à l'ordre six pour le cas du modèle cubique et jusqu'à l'ordre dix pour le modèle d'ordre cinq. Une simplification se basant sur l'hypothèse d'une entrée i.i.d. a été proposée par [129] pour le modèle cubique. Dans le cas des modèles de Volterra passe-bande, des simplifications ont aussi été proposées par [131] pour le modèle cubique et par [24] pour le modèle d'ordre cinq. Sous la même hypothèse d'entrée i.i.d. nous montrons comment l'estimation peut être simplifiée en utilisant le formalisme des polynômes orthogonaux.

Une telle approche a été considérée par Schetzen dans le cas d'un modèle de Volterra-Laguerre, pour une entrée gaussienne [121]. Les polynômes orthogonaux considérés dans ce cas précis sont les polynômes d'Hermite. Le modèle de Volterra étant poly-

nômial, le principe de la méthode est de le développer sur une base de polynômes orthogonaux.

Suivant la nature des signaux considérés, i.e. PAM, QAM ou PSK, et suivant l'ordre de non-linéarité du modèle de Volterra, différentes bases de polynômes orthogonaux peuvent être construites. En conséquence, nous traiterons chaque type de signal séparément. Nous restreignons notre étude au modèle d'ordre trois.

L'utilisation des polynômes orthogonaux n'est possible que lorsque le signal d'entrée est *i.i.d.*. Lorsque la complexité du modèle de Volterra est réduite à l'aide des BFOR, l'indépendance des entrées traitées, i.e. les entrées filtrées n'est plus de mise. Il est donc nécessaire de recourir à une procédure d'orthogonalisation. Dans le chapitre 2, nous avons considéré une structure orthogonale de type treillis pour des bases orthogonales de type Hambo. Il a été montré qu'une telle structure ne peut être générée dans le cas des filtres issus de la BFOR de Ninness et Gustafsson. Le recours à un module d'orthogonalisation a donc été suggéré. La mise en place d'un tel module sera considérée. Nous montrerons comment l'orthogonalisation des entrées permet d'améliorer le conditionnement de la matrice des moments de l'entrée et, en conséquence, d'accélérer la vitesse de convergence des algorithmes adaptatifs de type LMS.

Dans le cas particulier d'une entrée gaussienne, nous montrons qu'il est possible de coupler la réduction de complexité paramétrique à l'aide des BFOR et l'utilisation des bases de polynômes orthogonaux. D'un tel couplage résulte une estimation paramétrique sans inversion matricielle et dans le cas adaptatif une nette amélioration de la vitesse de convergence de l'algorithme LMS avec un surcoût modéré de la complexité de calcul.

4.2 Bases de polynômes orthogonaux multivariables

Dans cette section, nous allons exploiter les propriétés statistiques des entrées en vue de découpler l'estimation paramétrique du modèle de Volterra d'ordre trois. Pour le cas particulier des signaux PSK, nous indiquerons aussi une méthode d'estimation pour un modèle de Volterra d'ordre cinq de type passe-bande.

Avant tout, nous allons décrire quelques propriétés des signaux utilisés. L'énumération de ces propriétés est loin d'être exhaustive. Toutefois, elle regroupe toutes les propriétés nécessaires au développement de la méthode que nous proposons. Nous utiliserons des signaux de type PAM, PSK et QAM. Nous détaillerons davantage les propriétés des signaux complexes. Quant aux signaux PAM, la propriété fondamentale à retenir est qu'ils décrivent une distribution de probabilité symétrique.

4.2.1 Généralités sur les signaux complexes

Comme dans le cas des signaux réels, les signaux aléatoires complexes sont décrits par leurs propriétés moyennes en temps ou en fréquence. La description en temps se fait par les multicorrélations. Contrairement au cas réel, à un ordre donné, on peut définir différentes multicorrélations suivant le nombre de termes non conjugués ou conjugués pris en compte dans la définition de la multicorrélation. Une première définition est donnée en fonction des cumulants du signal [75] :

$$C_{p,q}(\underline{\tau}) = \text{Cum}[u(n), \dots, u(n + \tau_{p-1}), u^*(n + \tau_p), \dots, u^*(n + \tau_{p+q-1})]$$

avec $\underline{\tau} = [\tau_1 \dots \tau_{p+q-1}]$.

En notant $\mathbb{E}[\cdot]$ l'opérateur Espérance mathématique, une définition équivalente basée sur les moments [6] est : la multicorrélation $M_{p,q}$ est le moment d'ordre $p + q$ avec p termes non conjugués et q termes conjugués :

$$M_{p,q}(\underline{\tau}) = \mathbb{E} \left[\prod_{i=0}^{p-1} u(n + \tau_i) \prod_{j=p}^{p+q-1} u^*(n + \tau_j) \right], \quad \tau_0 = 0$$

Pour un signal i.i.d., la multicorrélation $M_{p,q}(\underline{\tau})$ s'annule sauf lorsque les instants τ_i sont égaux.

4.2.1.1 Circularité

La notion de circularité est issue de l'interprétation géométrique des variables aléatoires complexes. Une variable aléatoire complexe est circulaire si ses propriétés statistiques sont invariantes par rotation [75]. Une propriété importante découlant de cette observation est [6, 75, 28] :

les moments $M_{p,q}$ et les cumulants $C_{p,q}$ d'un signal circulaire sont nuls si $p \neq q$.

Au sens strict un signal aléatoire est circulaire si ses valeurs prises à n instants sont des variables aléatoires circulaires quel que soit le nombre d'instants n . Il s'en suit que les seules multicorrélations d'un signal aléatoire circulaire non nulles vérifient $p = q$:

$$M_{p,q}(\underline{\tau}) = \mathbb{E} \left[\prod_{i=0}^{p-1} u(n + \tau_i) \prod_{j=p}^{p+q-1} u^*(n + \tau_j) \right] = 0, \text{ si } p \neq q \quad \tau_0 = 0 \quad (4.1)$$

Circularité d'ordre α : On dit qu'un signal est circulaire d'ordre α si ses multicorrélations vérifient la propriété précédente jusqu'à l'ordre α

$$M_{p,q}(\underline{\tau}) = 0, \forall p, q \text{ tel que } p + q \leq \alpha \text{ et } p \neq q \quad (4.2)$$

4.2.1.2 Signaux PSK et QAM

En communication numérique, le signal émis u suit en général une loi de modulation connue. Plus couramment on rencontre des signaux de type BPSK, MSK, QPSK, 8-PSK et diverses QAM (4-QAM, 16-QAM, 64-QAM, 256-QAM). Les signaux M -PSK et M -QAM sont des signaux à alphabet fini, i.e. ils ne prennent qu'un ensemble fini de M valeurs. Cet ensemble de valeurs, appelé constellation, est :

– M -PSK

$$\mathcal{U} = \left\{ u_i = r \exp \left(j \frac{2\pi}{M} i \right), i = 0, 1, \dots, M-1 \right\}$$

– M -QAM

$$\mathcal{U} = \left\{ u_{i_1 i_2} = (2i_1 - \sqrt{M} - 1) + j(2i_2 - \sqrt{M} - 1), 1 \leq i_1, i_2 \leq \sqrt{M} \right\}$$

Notons que $\sqrt{M} = 2^L$ avec L un entier.

Une définition générale caractérisant l'ordre de circularité d'une variable aléatoire est [27] : une variable aléatoire est circulaire d'ordre $\alpha - 1$, $\alpha > 1$, si sa distribution est invariante par rotation d'angle $2\pi/\alpha$.

Les propriétés de circularité jouent un rôle important dans la classification des modulations utilisées en télécommunications. En particulier, les signaux de type M -PSK sont circulaires jusqu'à l'ordre $M - 1$ [75, 27]. Quant aux signaux M -QAM, $M = 2^{2L}$ avec L un entier, leur constellation est invariante par rotation d'angle $\pi/2$. Ils sont donc circulaires à l'ordre 3.

Une autre propriété mise en évidence par [24] est que les signaux de type M -PSK et M -QAM, uniformes i.i.d., $M \geq 4$, satisfont la relation suivante :

$$E[u^{2p+1}] = 0, p \geq 1 \quad (4.3)$$

Ceci vient du fait que la distribution de probabilité est symétrique. En conséquence, les moments d'ordre impair des signaux M -PSK et M -QAM sont tous nuls quelque soit l'ordre de leur circularité.

4.2.2 Identification d'un modèle de Volterra cubique

Considérons le modèle suivant :

$$\begin{aligned} y(n) = & \sum_{n_1=0}^{M-1} h_1(n_1)u(n-n_1) + \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} h_2(n_1, n_2)u(n-n_1)u(n-n_2) \\ & + \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \sum_{n_3=n_2}^{M-1} h_3(n_1, n_2, n_3)u(n-n_1)u(n-n_2)u(n-n_3) \end{aligned} \quad (4.4)$$

L'entrée u est supposée i.i.d. de type PAM dont l'alphabet contient au moins quatre valeurs distinctes.

L'entrée étant i.i.d, le modèle (4.4) peut être vu comme un polynôme de degré trois, à M variables indépendantes ($u(n), \dots, u(n - M + 1)$). Nous allons construire une base de polynômes orthogonaux multi-variables à partir de polynômes orthogonaux mono-variables.

4.2.2.1 Polynômes orthogonaux monovariabiles

Soit la variable aléatoire x admettant une distribution symétrique. Considérons l'espace de Hilbert $\mathbb{C}[X]$ des polynômes en x , auquel est associé le produit scalaire suivant :

Pour tous polynômes $A(x)$ et $B(x)$ appartenant à $\mathbb{C}[X]$

$$\langle A(x), B(x) \rangle = \mathbb{E}[A(x)B(x)] \quad (4.5)$$

où $\mathbb{E}$ est l'opérateur *espérance mathématique*.

Naturellement, l'ensemble $\{x^n\}_{n=0}^{\infty}$ est une base de cet espace. En appliquant une orthonormalisation de Gram-Schmidt à cette base, il est possible de générer une base orthonormale $\{P_n(x)\}_{n=0}^{\infty}$. Notre objectif étant de découpler un modèle cubique, nous considérerons le sous-espace de $\mathbb{C}[X]$ constitué de polynômes en x , de degré au plus égal à trois. Nous désignerons ce sous-espace par $\mathbb{C}_3[X]$. Une base de $\mathbb{C}_3[X]$ est $1, x, x^2, x^3$. Rappelons par ailleurs que x admettant une distribution de probabilité symétrique alors $\mu_n = \mathbb{E}[x^n] = 0$, si n est impair. A partir de la base $1, x, x^2, x^3$ de $\mathbb{C}_3[X]$, nous avons construit la base orthonormale $P_0(x), P_1(x), P_2(x), P_3(x)$ suivante (voir l'annexe C.1.1).

$$P_0(x) = 1 \quad (4.6)$$

$$P_1(x) = \frac{x}{\sqrt{\mu_2}} \quad (4.7)$$

$$P_2(x) = \frac{1}{\sqrt{\mu_4 - \mu_2^2}} (x^2 - \mu_2) \quad (4.8)$$

$$P_3(x) = \frac{1}{\sqrt{\mu_2^2 \mu_6 - \mu_4^2 \mu_2}} (\mu_2 x^3 - \mu_4 x) \quad (4.9)$$

Nous pouvons souligner les propriétés suivantes :

Propriété 4.1 Les polynômes orthogonaux $P_i(x)$, $i = 0, \dots, 3$ vérifient les relations suivantes :

1.

$$\langle P_m(x), P_n(x) \rangle = \mathbb{E}[P_m(x)P_n(x)] = \delta_{m,n} \quad (4.10)$$

2.

$$\mathbb{E}[P_n(x)] = 0, \quad n > 0 \quad (4.11)$$

3.

$$\mathbb{E}[x^m P_n(x)] = 0, \quad \text{si } m < n \quad (4.12)$$

Preuve : Les relations (4.10) et (4.11) sont respectivement l'expression de l'orthonormalité de la base et de l'orthogonalité des $P_n(x)$, $n > 0$ avec $P_0(x)$. Quant à (4.12), on peut remarquer que tout monôme x^m admet un développement sur la base des polynômes orthogonaux ; il peut donc s'écrire : $x^m = \sum_{i=0}^m b_i P_i(x)$. D'où :

$$\begin{aligned} \mathbb{E}[x^m P_n(x)] &= \sum_{i=0}^m b_i \mathbb{E}[P_i(x) P_n(x)] \\ &= \sum_{i=0}^m b_i \delta_{i,n} \\ &= \begin{cases} 0, & \text{si } m < n \\ b_n, & \text{si } m = n \end{cases} \end{aligned}$$

■

Remarque 4.1 Dans le cas où $\mu_4 = \mu_2^2$ et/ou $\mu_2 \mu_6 = \mu_4^2$, on prendra soin de ne pas normaliser les polynômes orthogonaux.

Application à l'estimation paramétrique d'un modèle cubique sans mémoire :
Considérons le modèle suivant :

$$y(n) = h_0 + h_1 u(n) + h_2 u^2(n) + h_3 u^3(n) = \theta^T U(n) = V(u(n)) \quad (4.13)$$

avec $\theta = (h_0 \ h_1 \ h_2 \ h_3)^T$ et $U(n) = (1 \ u(n) \ u^2(n) \ u^3(n))^T$. Le vecteur des paramètres optimaux au sens EQMM, lorsque d est la sortie réelle du système est :

$$\theta_{opt} = \Gamma_U^{-1} \mathfrak{C}_{dU}$$

avec :

$$\Gamma_U = \mathbb{E}[U(n)U(n)^T] = \begin{pmatrix} 1 & 0 & \mu_2 & 0 \\ 0 & \mu_2 & 0 & \mu_4 \\ \mu_2 & 0 & \mu_4 & 0 \\ 0 & \mu_4 & 0 & \mu_6 \end{pmatrix}, \quad \mathfrak{C}_{dU} = \mathbb{E}[d(n)U(n)]$$

La matrice Γ_U quoique creuse nécessite d'être inversée !

Considérons à présent le développement du polynôme $V(u(n))$ sur la base de $\mathbb{C}_3[u]$ que nous avons construite. Le modèle résultant est :

$$y(n) = f_0 P_0(u(n)) + f_1 P_1(u(n)) + f_2 P_2(u(n)) + f_3 P_3(u(n)) \quad (4.14)$$

avec

$$f_i = \mathbb{E}[V(u(n)) P_i(u(n))]$$

Posons $\check{\theta} = (f_0 \ f_1 \ f_2 \ f_3)^T$ et $\check{U}(n) = (P_0(u(n)) \ P_1(u(n)) \ P_2(u(n)) \ P_3(u(n)))^T$. Il va de soi que les paramètres optimaux au sens EQMM sont :

$$f_{i,opt} = \mathbb{E}[d(n)P_i(u)], \quad i = 0, \dots, 3 \quad (4.15)$$

4.2. Bases de polynômes orthogonaux multivariables

L'estimation des paramètres f_i du modèle (4.14) se fait donc sans aucun calcul matriciel et est complètement découlée.

Si on s'intéresse à retrouver les coefficients du modèle de départ, le calcul suivant est nécessaire :

$$\begin{aligned} f_i &= \mathbb{E}[V(u(n))P_i(u(n))] = \mathbb{E}\left[\sum_{j=0}^3 h_j u^j(n) P_i(u(n))\right] \\ &= \sum_{j=0}^3 h_j \mathbb{E}[u^j(n) P_i(u(n))] \\ &= \sum_{j=i}^3 h_j \mathbb{E}[u^j(n) P_i(u(n))] \end{aligned}$$

On obtient donc un système d'équations triangulaire qu'on résout par la technique de substitution. On a :

$$\begin{pmatrix} 1 & \mathbb{E}[u(n)P_0(u(n))] & \mathbb{E}[u^2(n)P_0(u(n))] & \mathbb{E}[u^3(n)P_0(u(n))] \\ 0 & \mathbb{E}[u(n)P_1(u(n))] & \mathbb{E}[u^2(n)P_1(u(n))] & \mathbb{E}[u^3(n)P_1(u(n))] \\ 0 & 0 & \mathbb{E}[u^2(n)P_2(u(n))] & \mathbb{E}[u^3(n)P_2(u(n))] \\ 0 & 0 & 0 & \mathbb{E}[u^3(n)P_3(u(n))] \end{pmatrix} \begin{pmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

En remplaçant les polynômes $P_i(u)$ par leurs expressions, on obtient :

$$\begin{pmatrix} 1 & 0 & \mu_2 & 0 \\ 0 & \sqrt{\mu_2} & 0 & \frac{\mu_4}{\sqrt{\mu_2}} \\ 0 & 0 & \sqrt{\mu_4 - \mu_2^2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{\mu_2 \mu_6 - \mu_2^2 \mu_4}}{\sqrt{\mu_2}} \end{pmatrix} \begin{pmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

d'où :

$$\begin{pmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\frac{\mu_2}{\sqrt{\mu_4 - \mu_2^2}} & 0 \\ 0 & \frac{1}{\sqrt{\mu_2}} & 0 & -\frac{\mu_4}{\sqrt{\mu_2^2 \mu_6 - \mu_2^2 \mu_4}} \\ 0 & 0 & \frac{1}{\sqrt{\mu_4 - \mu_2^2}} & 0 \\ 0 & 0 & 0 & \frac{\mu_2}{\sqrt{\mu_2^2 \mu_6 - \mu_2^2 \mu_4}} \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

4.2.2.2 Polynômes orthogonaux multivariables

Proposition 4.1 *Considérons deux variables indépendantes x_i et x_j admettant une même distribution de probabilité symétrique et P_l un ensemble de polynômes orthogonaux monovariables. Alors :*

$$\langle P_l(x_i), P_k(x_j) \rangle = \delta_{l,k} \delta_{i,j} \quad (4.16)$$

Preuve :

$$\langle P_l(x_i), P_k(x_j) \rangle = \mathbb{E}[P_l(x_i)P_k(x_j)] = \begin{cases} \mathbb{E}[P_l(x_i)]\mathbb{E}[P_k(x_j)], & \text{si } i \neq j \\ \mathbb{E}[P_l(x_i)P_k(x_i)], & \text{si } i = j \end{cases}$$

d'où :

$$\langle P_l(x_i), P_k(x_j) \rangle = \begin{cases} 0, & \text{si } i \neq j \\ \delta_{l,k}, & \text{sinon} \end{cases}$$

■

Nous pouvons en déduire que les polynômes $P_{n_1}(x_{n_1})P_{n_2}(x_{n_2})$ constituent une base orthogonale complète du sous-espace $\mathbb{C}_3[x_{n_1}, x_{n_2}]$ des polynômes à deux variables de degré au plus égal à trois. En effet :

$$\begin{aligned} \langle P_{m_1}(x_1)P_{m_2}(x_2), P_{n_1}(x_1)P_{n_2}(x_2) \rangle &= \mathbb{E}[P_{m_1}(x_1)P_{n_1}(x_1)]\mathbb{E}[P_{m_2}(x_2)P_{n_2}(x_2)] \\ &= \delta_{m_1, n_1}\delta_{m_2, n_2} \end{aligned} \quad (4.17)$$

Quant à la complétude, elle découle du théorème de Weierstrass à p dimensions qui stipule que l'ensemble des polynômes $\{x_1^{n_1}x_2^{n_2}\cdots x_p^{n_p}\}$ est complet en les p variables $x_1, x_2, \dots, x_p$ [121].

Nous pouvons aisément étendre cette théorie au cas général de p variables indépendantes suivant la même loi de probabilité.

Nous allons à présent considérer le cas du sous-espace $\mathbb{C}_3[x_1, x_2, \dots, x_p]$ des polynômes à p variables, $x_1, x_2, \dots, x_p$, de degré au plus égal à trois. Posons : $X = (x_1 x_2 \cdots x_p)^T$. Définissons à présent les polynômes orthogonaux multivariés nécessaires à la représentation de $\mathbb{C}_3[x_1, x_2, \dots, x_p]$. Ils sont notés $Q_{n_1, \dots, n_p}^{(p)}(X)$, i.e. $Q_{n_1, \dots, n_p}^{(p)}(X)$ est un polynôme de degré p en les variables $x_{n_1}, \dots, x_{n_p}$, $n_j = 1, \dots, p$, $j = 1, \dots, p$. On a :

– Degré zéro :

$$Q^{(0)}(X) = 1 \quad (4.18)$$

– Premier degré :

$$Q_{n_1}^{(1)}(X) = P_1(x_{n_1}) \quad (4.19)$$

– Second degré :

$$Q_{n_1, n_2}^{(2)}(X) = P_1(x_{n_1})P_1(x_{n_2}), \quad \text{si } n_1 \neq n_2 \quad (4.20)$$

$$Q_{n_1, n_1}^{(2)}(X) = P_2(x_{n_1}) \quad (4.21)$$

– Troisième degré :

$$Q_{n_1, n_2, n_3}^{(3)}(X) = P_1(x_{n_1})P_1(x_{n_2})P_1(x_{n_3}), \quad \text{si } n_1 \neq n_2 \neq n_3 \quad (4.22)$$

$$Q_{n_1, n_1, n_2}^{(3)}(X) = P_2(x_{n_1})P_1(x_{n_2}) \quad \text{si } n_1 \neq n_2 \quad (4.23)$$

$$Q_{n_1, n_1, n_1}^{(3)}(X) = P_3(x_{n_1}) \quad (4.24)$$

Remarque 4.2 Les polynômes Q sont appelés Q -polynômes par Schetzen [121] dans le cas des variables gaussiennes.

Propriété 4.2 Deux polynômes $Q(X)$ de même degré sont égaux si l'indice de l'un est obtenu en permutant les composantes de l'indice de l'autre¹.

Exemple 4.1

$$Q_{1,1,2}^{(3)}(X) = Q_{1,2,1}^{(3)}(X) = Q_{2,1,1}^{(3)}(X) = P_2(x_1)P_1(x_2)$$

Nous noterons $\alpha = (n_1, \dots, n_p)$.

Propriété 4.3 S'il n'existe aucune permutation π telle que $\alpha = \pi(\beta)$ alors :

$$\mathbb{E}[Q_\alpha^{(p)} Q_\beta^{(p)}] = 0 \quad (4.25)$$

Propriété 4.4

$$\mathbb{E}[Q_\alpha^{(p)} Q_\beta^{(q)}] = 0, \quad \text{si } p \neq q \quad (4.26)$$

Propriété 4.5 Soit $A(X)$ un polynôme multivariable en $x_1, \dots, x_p$. On a :

$$\mathbb{E}[A(X) Q_\alpha^{(p)}(X)] = 0, \quad \text{si } \deg(A) < p \quad (4.27)$$

Ces propriétés se démontrent en tenant compte de l'indépendance statistique des variables et de l'orthogonalité des polynômes à une dimension. Les polynômes Q constituent donc une base de $\mathbb{C}[x_1, \dots, x_p]$. Dans le cas de trois variables indépendantes, on a :

Degré	Polynômes orthogonaux multivariables
0	1
1	$Q_1^{(1)}(X), Q_2^{(1)}(X), Q_3^{(1)}(X)$
2	$Q_{1,1}^{(2)}(X), Q_{2,2}^{(2)}(X), Q_{3,3}^{(2)}(X),$ $Q_{1,2}^{(2)}(X), Q_{1,3}^{(2)}(X), Q_{2,3}^{(2)}(X),$
3	$Q_{1,1,1}^{(3)}(X), Q_{2,2,2}^{(3)}(X), Q_{3,3,3}^{(3)}(X),$ $Q_{1,2,3}^{(3)}(X), Q_{1,1,2}^{(3)}(X), Q_{1,1,3}^{(3)}(X),$ $Q_{2,2,3}^{(3)}(X), Q_{1,2,2}^{(3)}(X), Q_{1,3,3}^{(3)}(X), Q_{2,3,3}^{(3)}(X)$

TAB. 4.1 – Base orthonormale de $\mathbb{C}_3[x_1, x_2, x_3]$.

¹Le degré du polynôme P d'une variable x_{n_1} ne dépend que du nombre de fois que l'indice n_1 apparaît dans l'écriture de Q

Application à l'estimation paramétrique d'un modèle de Volterra cubique : Considérons le modèle (4.4). On définit par $U_1(n), U_2(n), U_3(n)$ les vecteurs de régression correspondant respectivement aux noyaux d'ordre 1, 2 et 3, de sorte qu'on a $U(n) = (U_1^T U_3^T U_2^T)^T$. Les coefficients des noyaux optimaux au sens EQMM sont tels que :

$$\theta = (\theta_1^T \theta_3^T \theta_2^T)^T = \mathbb{E}[U(n)U^T(n)]^{-1} \mathbb{E}[d(n)U(n)]$$

Sachant que u admet une distribution de probabilité symétrique, on a donc :

$$\mathbb{E}[U(n)U^T(n)] = \begin{pmatrix} \mathbb{E}[U_1(n)U_1^T(n)] & \mathbb{E}[U_1(n)U_3^T(n)] & 0 \\ \mathbb{E}[U_3(n)U_1^T(n)] & \mathbb{E}[U_3(n)U_3^T(n)] & 0 \\ 0 & 0 & \mathbb{E}[U_2(n)U_2^T(n)] \end{pmatrix}$$

La structure bloc-diagonale de la matrice d'autocorrélation, nous permet de déduire que l'estimation du noyau quadratique est découplée de celle des deux autres noyaux. Estimer les coefficients du noyau quadratique d'une part et les coefficients des noyaux linéaire et cubique de l'autre nécessite la coûteuse opération d'inversion matricielle.

Nous pouvons par ailleurs écrire $y(n) = V(U(n))$, où V est un polynôme multivariable en U . Il appartient donc au sous-espace $\mathbb{C}_3[u(n), \dots, u(n-M+1)]$. Son développement sur la base de polynômes orthogonaux que nous avons construite donne :

$$\begin{aligned} V(U(n)) &= \sum_{i=0}^{M-1} f_1(i) Q_i^{(1)}(U(n)) + \sum_{i_1=0}^{M-1} \sum_{i_2=i_1}^{M-1} f_2(i_1, i_2) Q_{i_1, i_2}^{(2)}(U(n)) \\ &\quad + \sum_{i_1=0}^{M-1} \sum_{i_2=i_1}^{M-1} \sum_{i_3=i_2}^{M-1} f_3(i_1, i_2, i_3) Q_{i_1, i_2, i_3}^{(3)}(U(n)) \end{aligned} \quad (4.28)$$

où

$$\begin{aligned} Q_i^{(1)}(U(n)) &= P_1(u(n-i)) \\ Q_{i_1, i_2}^{(2)}(U(n)) &= P_1(u(n-i_1))P_1(u(n-i_2)), \quad \text{si } i_1 \neq i_2 \\ Q_{i_1, i_1}^{(2)}(U(n)) &= P_2(u(n-i_1)) \\ Q_{i_1, i_2, i_3}^{(3)}(U(n)) &= P_1(u(n-i_1))P_1(u(n-i_2))P_1(u(n-i_3)), \quad \text{si } i_1 \neq i_2 \neq i_3 \\ Q_{i_1, i_1, i_2}^{(3)}(U(n)) &= P_2(u(n-i_1))P_1(u(n-i_2)) \quad \text{si } i_1 \neq i_2 \\ Q_{i_1, i_2, i_2}^{(3)}(U(n)) &= P_2(u(n-i_2))P_1(u(n-i_1)) \quad \text{si } i_1 \neq i_2 \\ Q_{i_1, i_1, i_1}^{(3)}(U(n)) &= P_3(u(n-i_1)) \end{aligned}$$

où les coefficients f_i sont donnés par

$$f_i(n_1, \dots, n_i) = \mathbb{E}[V(U(n))Q_{n_1, \dots, n_i}^{(i)}(U(n))] \quad (4.29)$$

La minimisation du critère $J_{EQM} = \mathbb{E}[d(n) - V(U(n))]$ donne, en tenant compte de l'orthogonalité de (4.28), les paramètres f_i optimaux suivants :

$$f_i(n_1, \dots, n_i) = \mathbb{E}[d(n)Q_{n_1, \dots, n_i}^{(i)}(U(n))] \quad (4.30)$$

L'estimation des paramètres du modèle orthogonalisé ne nécessite donc aucun calcul matriciel.

Si l'on désire retrouver les coefficients du modèle initial, la résolution d'un système d'équations triangulaire suffit. On peut écrire $V(U(n)) = \sum_{i=1}^3 V_i(U(n))$ où $V_i(U(n))$ est le polynôme multivariable en $U(n)$, de degré i , correspondant au terme homogène d'ordre i du modèle de Volterra. Tenant compte de (4.29) et de (4.27), nous obtenons le système d'équations triangulaire suivant :

$$f_i(n_1, \dots, n_i) = \sum_{j=i}^3 \mathbb{E}[V_j(U(n)) Q_{n_1, \dots, n_i}^{(i)}(U(n))] \quad (4.31)$$

Posons :

$$\kappa_{i,j} = \mathbb{E}[u^i(n) P_j(u(n))]$$

En développant les calculs, on montre que (voir l'annexe C.1.2) :

$$f_3(n_1, n_2, n_3) = \begin{cases} h_3(n_1, n_1, n_1) \kappa_{3,3}, & \text{si } n_1 = n_2 = n_3 \\ h_3(n_1, n_2, n_2) \kappa_{2,2} \kappa_{1,1}, & \text{si } n_1 \neq n_2 = n_3 \\ h_3(n_1, n_1, n_3) \kappa_{1,1} \kappa_{2,2}, & \text{si } n_1 = n_2 \neq n_3 \\ h_3(n_1, n_2, n_3) \kappa_{1,1}^3, & \text{sinon} \end{cases} \quad (4.32)$$

$$f_2(n_1, n_2) = \begin{cases} h_2(n_1, n_1) \kappa_{2,2}, & \text{si } n_1 = n_2 \\ h_2(n_1, n_2) \kappa_{1,1}^2, & \text{sinon} \end{cases} \quad (4.33)$$

$$f_1(n_1) = h_1(n_1) \kappa_{1,1} + h_3(n_1, n_1, n_1) \kappa_{3,1} + \mu_2 \kappa_{1,1} \left(\sum_{i < n_1}^{M-1} h_3(i, i, n_1) + \sum_{i > n_1}^{M-1} h_3(n_1, i, i) \right) \quad (4.34)$$

où :

$$\begin{aligned} \kappa_{1,1} &= \sqrt{\mu_2}; & \kappa_{3,1} &= \frac{\mu_4}{\sqrt{\mu_2}} \\ \kappa_{2,2} &= \sqrt{\mu_4 - \mu_2^2}; & \kappa_{3,3} &= \sqrt{\mu_6 - \frac{\mu_4^2}{\mu_2}} \end{aligned}$$

Par suite, nous avons :

$$\begin{aligned}
 h_3(n_1, n_1, n_1) &= \frac{f_3(n_1, n_1, n_1)}{\sqrt{\mu_6 - \frac{\mu_4^2}{\mu_2}}} \\
 h_3(n_1, n_1, n_3) &= \frac{f_3(n_1, n_1, n_3)}{\sqrt{\mu_4 \mu_2 - \mu_2^3}} \\
 h_3(n_1, n_2, n_2) &= \frac{f_3(n_1, n_2, n_2)}{\sqrt{\mu_4 \mu_2 - \mu_2^3}} \\
 h_3(n_1, n_2, n_3) &= \frac{f_3(n_1, n_2, n_3)}{\sqrt{\mu_2^3}} \\
 h_2(n_1, n_1) &= \frac{f_2(n_1, n_1)}{\sqrt{\mu_4 - \mu_2^2}} \\
 h_2(n_1, n_2) &= \frac{f_2(n_1, n_2)}{\mu_2}
 \end{aligned} \tag{4.35}$$

$$\begin{aligned}
 h_1(n_1) &= \frac{f_1(n_1)}{\sqrt{\mu_2}} - \frac{\mu_4}{\sqrt{\mu_6 \mu_2^2 - \mu_2 \mu_4^2}} f_3(n_1, n_1, n_1) \\
 &\quad - \frac{\mu_2}{\sqrt{\mu_4 \mu_2 - \mu_2^3}} \left(\sum_{i < n_1}^{M-1} f_3(i, i, n_1) + \sum_{i > n_1}^{M-1} f_3(n_1, i, i) \right)
 \end{aligned} \tag{4.36}$$

4.2.3 Identification d'un modèle de Volterra passe-bande d'ordre trois à l'aide de signaux QAM et PSK

Pour un système de communication à bande étroite, la relation entre les signaux complexes d'entrée et de sortie peut être modélisée par le filtre de Volterra suivant :

$$\begin{aligned}
 y(n) &= \sum_{n_1=0}^{M-1} h_1(n_1) u(n - n_1) \\
 &\quad + \sum_{n_1=0}^{M-1} \sum_{n_2=n_1}^{M-1} \sum_{n_3=0}^{M-1} h_3(n_1, n_2, n_3) u(n - n_1) u(n - n_2) u^*(n - n_3)
 \end{aligned} \tag{4.37}$$

où les coefficients $h_1(n_1)$ et $h_3(n_1, n_2, n_3)$ ont des valeurs complexes [8].

4.2.3.1 Polynômes orthogonaux monovariables

Le modèle ci-dessus est un polynôme en $u(n), \dots, u(n - n_1), \dots, u(n - M + 1)$. Pour n_1 fixé, posons $x = u(n - n_1)$. La relation (4.37) peut alors s'écrire telle que :

$$\begin{aligned}
 y(n) = & \alpha(u(n), \dots, u(n - n_1 - 1), u(n - n_1 + 1), \dots, u(n - M + 1)) \\
 & + h_1(n_1)x + h_3(n_1, n_1, n_1)x^2x^* + xx^* \sum_{i=0}^{n_1-1} h_3(i, n_1, n_1)u(n - i) \\
 & + x^2 \sum_{i=0, i \neq n_1}^{M-1} h_3(n_1, n_1, i)u^*(n - i) + xx^* \sum_{i=n_1+1}^{M-1} h_3(n_1, i, n_1)u(n - i) \\
 & + x \sum_{i=0}^{n_1-1} \sum_{j=0, j \neq n_1}^{M-1} h_3(i, n_1, j)u(n - i)u^*(n - j) \\
 & + x \sum_{i=n_1+1}^{M-1} \sum_{j=0, j \neq n_1}^{M-1} h_3(n_1, i, j)u(n - i)u^*(n - j) \\
 & + x^* \sum_{i=0, i \neq n_1}^{M-1} \sum_{j=i, j \neq n_1}^{M-1} h_3(i, j, n_1)u(n - i)u(n - j)
 \end{aligned} \tag{4.38}$$

En supposant que seule x est variable, nous obtenons une famille de polynômes en x générée par la base suivante :

$$1, x, x^*, x^2, xx^*, x^2x^* \tag{4.39}$$

Cette base génère un sous ensemble de l'espace de Hilbert $\mathbb{C}_3[x]$ des polynômes en x de degré inférieur ou égal à trois auquel est associé le produit scalaire suivant :

Pour tous polynômes $A(x)$ et $B(x)$ appartenant à $\mathbb{C}_3[x]$

$$\langle A(x), B(x) \rangle = \mathbb{E}[A(x)B^*(x)] \tag{4.40}$$

A partir de la base décrite ci-dessus, nous allons construire une base orthonormale par orthonormalisation de Gram-Schmidt. Nous obtenons une base orthonormale dont les éléments $P_{i,j}(x)$ sont des polynômes en x et en x^* de degré total $i + j$, i et j étant respectivement le degré associé à x et à x^* . Nous notons par ailleurs $\mu_{i,j}$ le moment d'ordre $i + j$ défini par :

$$\mu_{i,j} = \mathbb{E}[x^i x^{*j}]$$

Les éléments de la base orthogonale sont alors (Voir l'annexe C.2.1 pour le détail de la construction) :

$$P_{0,0}(x) = 1 \tag{4.41}$$

$$P_{1,0}(x) = \frac{x}{\sqrt{\mu_{1,1}}} \tag{4.42}$$

$$P_{0,1}(x) = \frac{x^*}{\sqrt{\mu_{1,1}}} \tag{4.43}$$

$$P_{2,0}(x) = \frac{x^2}{\sqrt{\mu_{2,2}}} \quad (4.44)$$

$$P_{1,1}(x) = \frac{xx^* - \mu_{1,1}}{\sqrt{\mu_{2,2} - \mu_{1,1}^2}} \quad (4.45)$$

$$P_{2,1}(x) = \frac{\mu_{1,1}x^2x^* - \mu_{2,2}x}{\sqrt{\mu_{3,3}\mu_{1,1}^2 - \mu_{1,1}\mu_{2,2}^2}} \quad (4.46)$$

Remarque 4.3 On notera l'égalité suivante :

$$P_{1,0}(x) = P_{0,1}(x^*)$$

Remarque 4.4 Comme indiqué dans le cas des séquences PAM, nous devons nous assurer que la séquence considérée a ses moments qui soient tels que $\mu_{3,3}\mu_{1,1}^2 - \mu_{1,1}\mu_{2,2}^2 \neq 0$ et $\mu_{2,2} - \mu_{1,1}^2 \neq 0$. C'est le cas d'une séquence QAM - 4

Propriétés : Avant de construire les bases de polynômes multivariées, nous allons énoncer quelques propriétés des polynômes monovariés que nous venons de construire.

Propriété 4.6 Les polynômes orthogonaux $P_{i,j}(x)$, $i = 0, 1, 2$, $j = 0, 1$, vérifient les relations suivantes :

1.

$$\langle P_{i_1,j_1}(x), P_{i_2,j_2}(x) \rangle = \mathbb{E} [P_{i_1,j_1}(x)P_{i_2,j_2}^*(x)] = \delta_{i_1,i_2} \delta_{j_1,j_2} \quad (4.47)$$

2.

$$\mathbb{E} [P_{i,j}(x)] = 0, \quad i + j > 0 \quad (4.48)$$

3.

$$\mathbb{E} [x^m x^{*n} P_{i,j}^*(x)] = 0, \quad \text{si } m < i \text{ et } n < j \quad (4.49)$$

Preuve : Les relations (4.47) et (4.48) sont respectivement l'expression de l'orthonormalité de la base et de l'orthogonalité des $P_{i,j}(x)$, $i + j > 0$ avec $P_{0,0}(x)$. Quant à (4.49), on peut remarquer que tout monôme $x^m x^{*n}$ admet un développement sur la base des polynômes orthogonaux ; il peut donc s'écrire : $x^m x^{*n} = \sum_{i_1=0}^m \sum_{j_1=0}^n b_{i_1,j_1} P_{i_1,j_1}(x)$. D'où :

$$\begin{aligned} \mathbb{E} [x^m x^{*n} P_{i,j}^*(x)] &= \sum_{i_1=0}^m \sum_{j_1=0}^n b_{i_1,j_1} \mathbb{E} [P_{i_1,j_1}(x) P_{i,j}^*(x)] \\ &= \sum_{i_1=0}^m \sum_{j_1=0}^n b_{i_1,j_1} \delta_{i_1,i} \delta_{j_1,j} \\ &= \begin{cases} b_{i,j}, & \text{si } 0 \leq i \leq m \text{ et } 0 \leq j \leq n \\ 0, & \text{sinon} \end{cases} \end{aligned}$$

■

4.2.3.2 Polynômes orthogonaux multivariables

Proposition 4.2 *Considérons deux variables indépendantes x_r et x_s , de type QAM, suivant une distribution uniforme. Alors :*

$$\langle P_{l_i, k_i}(x_r), P_{l_j, k_j}(x_s) \rangle = \delta_{l_i, l_j} \delta_{k_i, k_j} \quad (4.50)$$

Preuve :

$$\begin{aligned} \langle P_{l_i, k_i}(x_r), P_{l_j, k_j}(x_s) \rangle &= \mathbb{E} [P_{l_i, k_i}(x_r) P_{l_j, k_j}^*(x_s)] \\ &= \begin{cases} \mathbb{E} [P_{l_i, k_i}(x_r)] \mathbb{E} [P_{l_j, k_j}^*(x_s)], & \text{si } r \neq s \\ \mathbb{E} [P_{l_i, k_i}(x_r) P_{l_j, k_j}^*(x_r)], & \text{sinon} \end{cases} \end{aligned}$$

d'où :

$$\langle P_{l_i, k_i}(x_r), P_{l_j, k_j}(x_s) \rangle = \begin{cases} 0, & \text{si } r \neq s \\ \delta_{l_i, l_j} \delta_{k_i, k_j}, & \text{sinon} \end{cases}$$

■

Nous pouvons en déduire que les polynômes $P_{l_1, j_1}(x_{n_1}) P_{l_2, j_2}(x_{n_2})$ constituent une base orthonormale d'un sous-espace $\mathbb{C}_3[x_{n_1}, x_{n_2}]$ des polynômes à deux variables de degré au plus égal à trois.

Nous pouvons aisément étendre cette théorie au cas général de M variables indépendantes suivant la même loi de probabilité. Pour ce faire, nous allons définir des polynômes orthogonaux multivariables pour des variables de type QAM.

Considérons une séquence i.i.d à M variables $u(n), \dots, u(n-M+1)$. La base de polynômes orthogonaux multivariables nécessaires à la représentation de (4.37) est donnée par :

– Premier degré :

$$Q_i^{(1)}(U(n)) = P_{1,0}(u(n-i))$$

– Troisième degré :

$$\begin{aligned} Q_{i,j,k}^{(3)}(U(n)) &= P_{1,0}(u(n-i)) P_{1,0}(u(n-j)) P_{1,0}(u^*(n-k)), \quad i \neq j \neq k \\ Q_{i,i,k}^{(3)}(U(n)) &= P_{2,0}(u(n-i)) P_{1,0}(u^*(n-k)), \quad i \neq k \\ Q_{i,k,k}^{(3)}(U(n)) &= P_{1,0}(u(n-i)) P_{1,1}(u(n-k)), \quad i \neq k \\ Q_{i,i,i}^{(3)}(U(n)) &= P_{2,1}(u(n-i)) \end{aligned}$$

Propriétés : Les polynômes multivariables ainsi construits vérifient les propriétés suivantes :

Pour $U(n) = (u(n), \dots, u(n-n_1), \dots, u(n-M+1))$ et en posant $\alpha = (n_1, \dots, n_p)$ et $\beta = (m_1, \dots, m_p)$, on a :

Propriété 4.7

$$\mathbb{E}[Q_\alpha^{(p)}(U(n))Q_\beta^{(q)}(U(n))] = 0, \quad \text{si } p \neq q \quad (4.51)$$

Propriété 4.8 Soit $A(U(n))$ un polynôme multivariable en $u(n), \dots, u(n-M+1)$. On a :

$$\mathbb{E}[A(U(n))Q_\alpha^{(p)}(U(n))] = 0, \quad \text{si } \deg(A) < p \quad (4.52)$$

4.2.3.3 Application à l'identification d'un modèle de Volterra passe-bande d'ordre trois

Rappelons que le modèle (4.37) peut s'écrire $y(n) = V(U(n))$ où $V(U(n))$ est un polynôme multivariable en $u(n), \dots, u(n-M+1)$. Ce polynôme se décompose sur la base que nous avons construite de la manière suivante :

$$V(U(n)) = \sum_{i=0}^{M-1} f_1(i)Q_i^{(1)}(U(n)) + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=0}^{M-1} f_3(i, j, k)Q_{i,j,k}^{(3)}(U(n)) \quad (4.53)$$

Les paramètres optimaux au sens EQMM sont alors donnés par :

$$f_1(i) = \mathbb{E}[y^*(n)Q_i^{(1)}(U(n))] \quad (4.54)$$

$$f_3(i, j, k) = \mathbb{E}[y^*(n)Q_{i,j,k}^{(3)}(U(n))] \quad (4.55)$$

Si l'on désire obtenir les coefficients des noyaux de Volterra du modèle non orthogonal, un système d'équations triangulaire est à résoudre. En effet, on peut remarquer qu'on peut écrire $V(U(n)) = \sum_{i=0}^1 V_{2i+1}(U(n))$ où $V_i(U(n))$ est le polynôme multivariable en $U(n)$, de degré i , correspondant au terme homogène d'ordre i du modèle de Volterra. Sachant que $f_i(n_1, \dots, n_i) = \mathbb{E}[Q_{n_1, \dots, n_i}^{(i)}(U(n))V^*(U(n))]$ et tenant compte de (4.52), nous obtenons le système d'équations triangulaire suivant :

$$f_{2i+1}(n_1, \dots, n_{2i+1}) = \sum_{j=i}^1 \mathbb{E}[Q_{n_1, \dots, n_{2i+1}}^{(2i+1)}(U(n))V_{2j+1}^*(U(n))] \quad (4.56)$$

En particulier, on a :

$$\begin{aligned} f_3(n_1, n_2, n_3) &= \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=0}^{M-1} h_3^*(i, j, k) \\ &\quad \times \mathbb{E}[Q_{n_1, n_2, n_3}^{(3)}(U(n))u^*(n-i)u^*(n-j)u(n-k)] \\ f_1(n_1) &= \sum_{i=0}^{M-1} h_1^*(i) \mathbb{E}[Q_{n_1}^{(1)}(U(n))u^*(n-i)] \\ &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=0}^{M-1} h_3^*(i, j, k) \mathbb{E}[Q_{n_1}^{(1)}(U(n))u^*(n-i)u^*(n-j)u(n-k)] \end{aligned} \quad (4.57)$$

Noyaux de Volterra optimaux pour une entrée QAM : Nous avons les relations suivantes ² :

$$\begin{aligned}
 h_3(n_1, n_2, n_3) &= \frac{f_3^*(n_1, n_2, n_3)}{\sqrt{\mu_{1,1}^3}}, n_1 \neq n_2 \neq n_3 \\
 h_3(n_1, n_1, n_2) &= \frac{f_3^*(n_1, n_1, n_2)}{\sqrt{\mu_{1,1}\mu_{2,2}}}, n_1 \neq n_2 \\
 h_3(n_1, n_2, n_2) &= \frac{f_3^*(n_1, n_2, n_2)}{\sqrt{\mu_{1,1}\mu_{2,2}-\mu_{1,1}^3}}, n_1 \neq n_2 \\
 h_3(n_1, n_2, n_1) &= \frac{f_3^*(n_1, n_2, n_1)}{\sqrt{\mu_{1,1}\mu_{2,2}-\mu_{1,1}^3}}, n_1 \neq n_2 \\
 h_3(n_1, n_1, n_1) &= \frac{\mu_{1,1}f_3^*(n_1, n_1, n_1)}{\sqrt{\mu_{1,1}^2\mu_{3,3}-\mu_{1,1}\mu_{2,2}^2}}
 \end{aligned} \tag{4.58}$$

et

$$\begin{aligned}
 h_1(n_1) &= \frac{f_1^*(n_1)}{\sqrt{\mu_{1,1}}} - \frac{\mu_{2,2}f_3^*(n_1, n_1, n_1)}{\sqrt{\mu_{1,1}^2\mu_{3,3}-\mu_{1,1}\mu_{2,2}^2}} \\
 &\quad - \frac{\mu_{1,1}}{\sqrt{\mu_{1,1}\mu_{2,2}-\mu_{1,1}^2}} \left(\sum_{i>n_1} f_3^*(n_1, i, i) + \sum_{i<n_1} f_3^*(i, n_1, i) \right)
 \end{aligned} \tag{4.59}$$

Noyaux de Volterra optimaux pour une entrée de type PSK : Dans le cas d'une entrée de type PSK, certains coefficients du noyau d'ordre trois dégénèrent en des coefficients du noyau du premier ordre. Ceci vient du fait que les signaux PSK sont de module constant. Dans l'expression (4.37), chaque coefficient $h_3(n_1, n_2, n_3)$ pondère un produit de type $u(n-n_1)u(n-n_2)u^*(n-n_3)$. Lorsque le décalage n_3 du terme conjugué est égal à au moins un décalage d'un terme non conjugué, le terme non-linéaire $u(n-n_1)u(n-n_2)u^*(n-n_3)$ dégénère en un terme linéaire. On a en effet :

$$u(n-n_1)u(n-n_2)u^*(n-n_3) = \begin{cases} \alpha u(n-n_1) & \text{si } n_2 = n_3 \\ \alpha u(n-n_1) & \text{si } n_1 = n_2 = n_3 \\ \alpha u(n-n_2) & \text{si } n_1 = n_3 \\ u^2(n-n_1)u^*(n-n_3) & \text{si } n_1 = n_2 \\ u(n-n_1)u(n-n_2)u^*(n-n_3) & \text{sinon} \end{cases}$$

où α est le carré du module de la séquence PSK. En conséquence, les seuls coefficients du noyau du troisième ordre ne dégénérant pas en coefficients du noyau du premier ordre sont de type $h_3(n_1, n_2, n_3)$ et $h_3(n_1, n_1, n_3)$, $n_1 \neq n_2 \neq n_3$. Il a été montré dans [129] que l'estimation des paramètres, dans ce cas, est complètement découplée. Les noyaux optimaux sont donnés par :

$$\begin{aligned}
 h_3(n_1, n_2, n_3) &= \frac{\mathbb{E}[d(n)u^*(n-n_1)u^*(n-n_2)u(n-n_3)]}{\mu_{1,1}^3} \\
 h_3(n_1, n_1, n_2) &= \frac{\mathbb{E}[d(n)u^*(n-n_1)u^*(n-n_1)u(n-n_2)]}{\mu_{1,1}\mu_{2,2}}
 \end{aligned} \tag{4.60}$$

²Le calcul détaillé est donné dans l'annexe C.2.2

et

$$h_1(n_1) = \frac{\mathbb{E}[d(n)u^*(n-n_1)]}{\mu_{1,1}} \quad (4.61)$$

Il est à noter que dans le cas des modèles de Volterra passe-bande d'ordre cinq, il n'y a plus de découplage [24].

4.2.4 Estimation adaptative basée sur le formalisme des polynômes orthogonaux

Considérons un modèle de Volterra cubique et sa représentation dans une base de polynômes orthogonaux multivariables, respectivement donnés par :

$$\begin{aligned} y(n) &= \sum_{p=1}^3 \sum_{n_1=0}^{M-1} \cdots \sum_{n_p=n_{p-1}}^{M-1} h_p(n_1, \dots, n_p) \prod_{j=1}^p u(n-n_j) = \Theta^T U(n) \\ &= \sum_{p=1}^3 \sum_{i_1=0}^{M-1} \cdots \sum_{i_p=i_{p-1}}^{M-1} f_p(i_1, \dots, i_p) Q_{i_1, \dots, i_p}^{(p)}(U_1) = \bar{\Theta}^T Q(n) \end{aligned}$$

avec

$$\bar{\Theta} = (f_1(0), \dots, f_3(M-1, M-1, M-1))^T$$

et

$$Q(n) = \left(Q_0^{(1)}(U_1(n)), \dots, Q_{M-1, M-1, M-1}^{(3)}(U_1(n)) \right)^T$$

où les $Q_{i_1, \dots, i_p}^{(p)}(U_1(n))$ sont des éléments de la base de polynômes orthogonaux multivariables. On a donc $\mathbb{E}[Q(n)Q^T(n)] = \mathbf{I}$.

Supposons qu'on veuille identifier un système non-linéaire à l'aide d'un filtre de Volterra. En utilisant l'algorithme LMS pour l'estimation adaptative des paramètres du filtre de Volterra, on s'attend à avoir une convergence lente. La vitesse de convergence de l'algorithme LMS dépend fortement du conditionnement de la matrice des moments de l'entrée. En tenant compte de la représentation du filtre de Volterra dans la base des polynômes orthogonaux, la matrice des moments est alors, théoriquement, égale à la matrice unité. Son nombre condition est égal à un. Ainsi, en estimant les paramètres du système avec la représentation sur la base de polynômes orthogonaux, on peut s'attendre à une amélioration de la vitesse de convergence de l'algorithme LMS. La structure d'estimation basée sur ce principe est décrite par la figure 4.1.

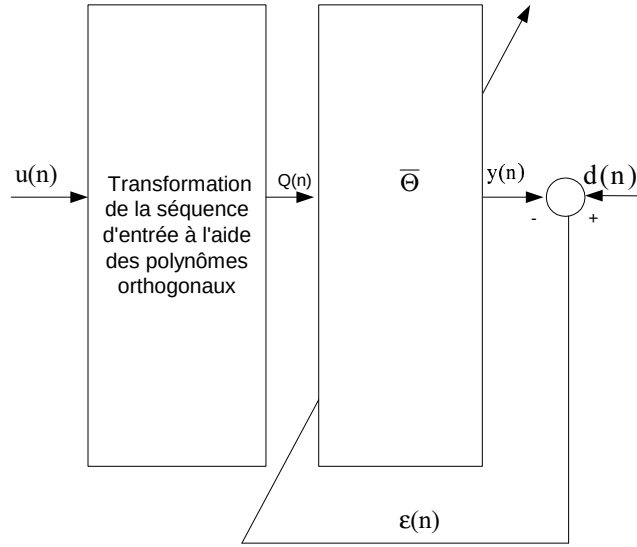


FIG. 4.1 – Estimation adaptative basée sur le formalisme des polynômes orthogonaux

4.2.5 Simulations

4.2.5.1 Identification d'un modèle cubique sans mémoire à l'aide de signaux PAM

Considérons le système sans mémoire décrit par la relation suivante :

$$y(n) = 1.3u(n) - 0.5u^2(n) + 0.16u^3(n) + w(n)$$

L'entrée est une séquence PAM-4, i.i.d, centrée et de variance unité. Les résultats ci-dessous sont obtenus en moyennant vingt expériences indépendantes pour chaque nombre de données N et chaque rapport signal à bruit. Le bruit additif $w(n)$ est blanc et gaussien.

Comme on peut le remarquer sur la figure 4.2 deux types de bruit sont mis en évidence : le bruit additif gaussien et l'erreur d'estimation des moments due au nombre de données utilisées pour l'estimation des moments nécessaires à notre algorithme. En effet, la procédure d'orthonormalisation nécessite des statistiques exactes. Or, celles-ci ne sont approchées que lorsque le nombre de données est important. Il est à noter que le défaut d'estimation de ces statistiques est noyé dans le bruit additif lorsque celui-ci est important. C'est ce qui explique que pour un rapport signal à bruit inférieur à $20dB$ les performances ne s'améliorent que très peu malgré l'augmentation significative du nombre de données. Par contre pour un bruit plus faible l'effet d'augmenter le nombre de données se fait davantage ressentir sur l'estimation des statistiques et donc sur l'estimation des paramètres du modèle.

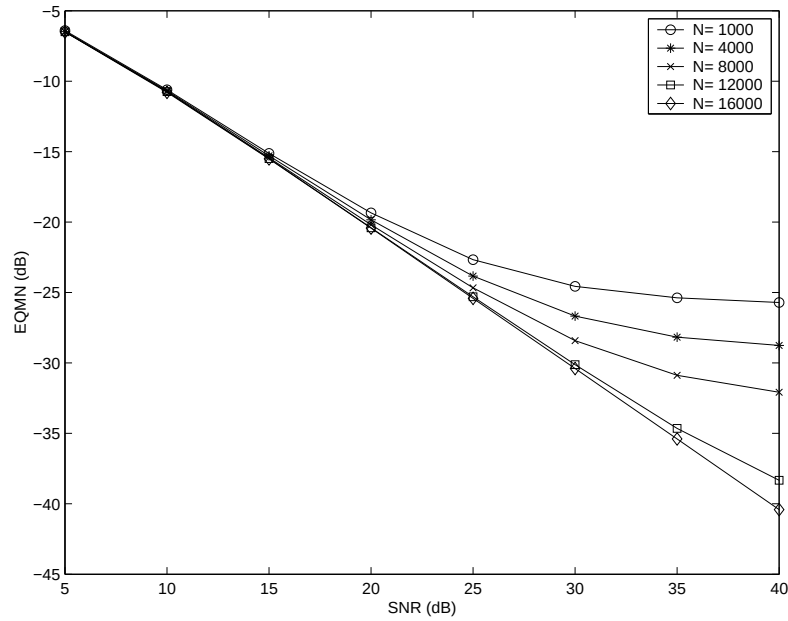


FIG. 4.2 – EQMN en fonction du rapport signal à bruit pour un modèle cubique sans mémoire

Dans le tableau 4.2 nous comparons l'erreur quadratique moyenne normalisée résultant de la méthode proposée dans la section 4.2.2 avec celle résultant d'une estimation au sens EQMM classique, i.e. nécessitant l'inversion de la matrice d'autocorrélation. Nous pouvons constater que les performances sont identiques. Notre méthode présente

SNR (dB)	5	15	25	35
Méthode Classique (dB)	-6.474	-15.519	-25.340	-35.401
Méthode Orthogonale (dB)	-6.493	-15.490	-25.300	-34.649

TAB. 4.2 – Comparaison avec la méthode EQM classique dans le cas d'un modèle cubique sans mémoire ($N = 12000$)

l'avantage d'être moins coûteuse en temps de calcul, puisqu'aucune inversion matricielle n'est requise.

4.2.5.2 Identification d'un modèle cubique à l'aide de signaux PAM

Considérons le modèle de Volterra cubique représentant un canal non-linéaire formé par la mise en cascade d'un canal linéaire et d'une partie non-linéaire respectivement représentées par : $H(z) = 0.35 + 0.87z^{-1} + 0.35z^{-2}$ et $P(u) = 0.99u - 0.35u^3$. Nous

4.2. Bases de polynômes orthogonaux multivariables

considérons l'identification de ce canal à l'aide d'une séquence PAM-4, *i.i.d.* de variance unité. Pour chaque niveau de bruit, 50 expériences indépendantes sont menées. Les résultats obtenus sont donnés par la figure 4.3.

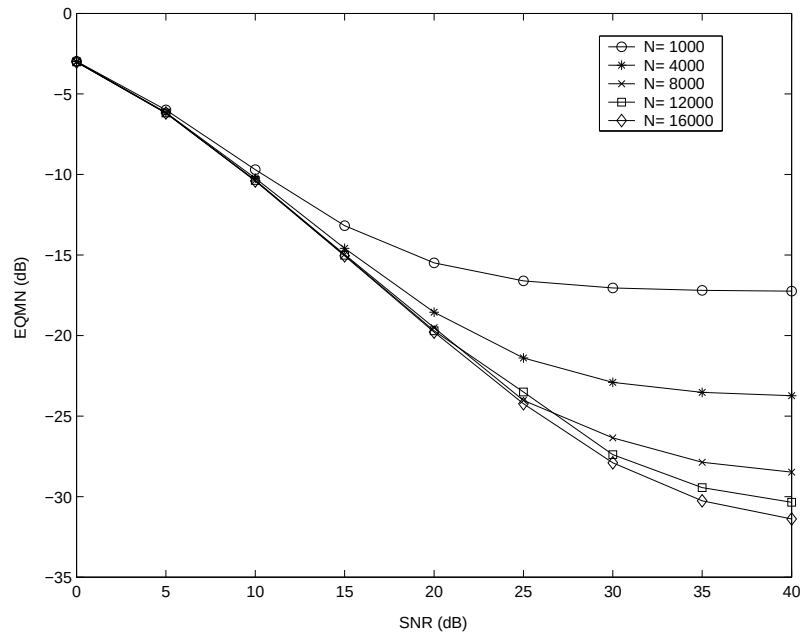


FIG. 4.3 – EQMN en fonction du rapport signal à bruit

Dans le tableau 4.3, nous montrons une comparaison avec la méthode d'estimation classique au sens EQMM et la méthode proposée par [131] dont un résumé est donné en annexe D.1.

SNR (dB)	5	15	25	35
Méthode Classique (dB)	-6.196	-15.148	-25.016	-34.997
Méthode de Tseng et Powers (dB)	-6.186	-15.113	-24.686	-32.536
Méthode Orthogonale (dB)	-6.180	-15.105	-24.228	-30.258

TAB. 4.3 – EQMN pour une entrée PAM-4 ($N = 16000$)

Méthode	classique	orthogonale
Nombre condition	70.416	1.095

TAB. 4.4 – Conditionnement de la matrice des moments de l'entrée d'un modèle de Volterra cubique pour une entrée PAM-4

Nous pouvons constater que les résultats donnés par les trois méthodes sont similaires. Il est à noter que la méthode classique requiert l'inversion d'une matrice carrée d'ordre 13, ce qui équivaut à un coût de calcul d'ordre $O(13^3)$ tandis que les deux autres méthodes n'en nécessitent point. Par ailleurs l'avantage de la méthode orthogonale est de donner des expressions analytiques des paramètres optimaux plus simples que celles de la méthode [131]. Ces expressions étant données par les équations (4.30). A titre indicatif nous avons estimé la matrice des moments de l'entrée puis évalué son nombre condition. Comme l'atteste le tableau 4.4, la matrice des moments est mieux conditionnée dans la méthode orthogonale.

Nous pouvons noter une nette amélioration du conditionnement numérique de la matrice des moments. Voyons à présent comment cela se traduit dans une estimation adaptative à l'aide de l'algorithme LMS. Le rapport signal à bruit est fixé à 30 dB, une évaluation des performances, de type Monte Carlo, est réalisée avec 100 expériences indépendantes. Le pas d'adaptation de l'algorithme LMS est fixé à $\mu = 0.08$.

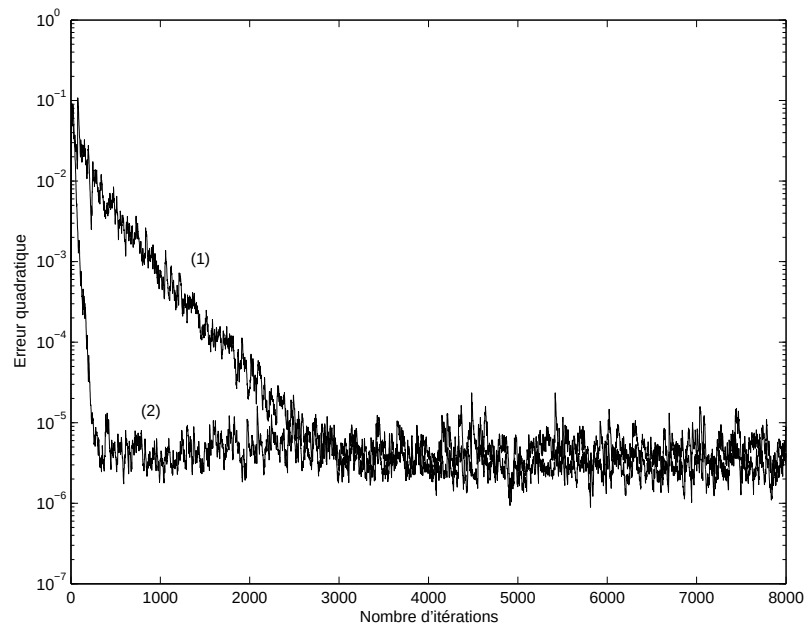


FIG. 4.4 – Erreur quadratique d'estimation d'un modèle cubique : (1) LMS, (2) LMS basé sur le formalisme des polynômes orthogonaux

Comme prévu par la théorie, l'algorithme LMS utilisant des données transformées par les polynômes orthogonaux, converge bien plus rapidement que l'algorithme standard.

4.2.5.3 Identification d'un modèle de Volterra passe-bande d'ordre trois à l'aide de signaux QAM

Nous considérons l'identification d'un canal satellitaire en considérant les mesures de coefficients des noyaux de Volterra effectuées par [8]. Les seuls coefficients non nuls du modèle d'ordre trois sont donnés par le tableau 4.5 :

Noyau d'ordre 1	Noyau d'ordre 3	
$h_1(0) = 3.4 + 0.381j$	$h_3(0,0,0) = -4.926 + 1.741j$	$h_3(0,1,1) = -0.230 + 0.081j$
$h_1(1) = 0.052 + 0.006j$	$h_3(0,0,1) = 0.194 - 0.068j$	$h_3(0,2,0) = -0.056 + 0.020j$
$h_1(2) = -0.048 - 0.005j$	$h_3(0,0,3) = -0.192 + 0.068j$	$h_3(0,3,0) = -0.384 + 0.136j$
$h_1(3) = 0.178 + 0.020j$	$h_3(0,1,0) = 0.388 - 0.137j$	$h_3(0,3,3) = -0.033 + 0.029j$
	$h_3(1,1,0) = -0.115 + 0.041j$	$h_3(1,1,1) = -0.105 + 0.037j$
	$h_3(2,2,2) = 0.074 - 0.026j$	$h_3(3,3,0) = -0.041 + 0.015j$

TAB. 4.5 – Coefficients du modèle de Volterra d'ordre trois représentant un canal satellitaire

Nous utilisons une entrée de type QAM-16, *i.i.d.* Pour chaque niveau de bruit, 50 expériences indépendantes sont menées. Les résultats obtenus sont donnés par la figure 4.5.

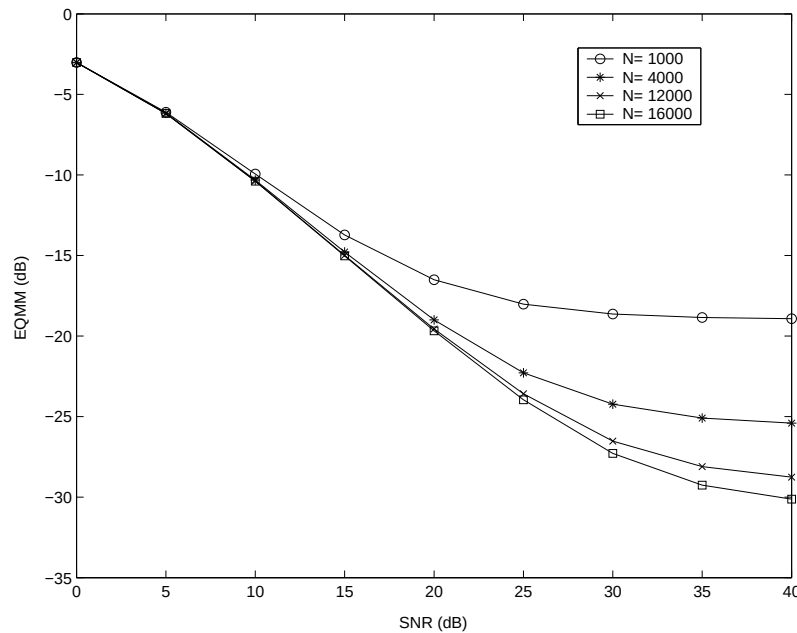


FIG. 4.5 – EQMN en fonction du rapport signal à bruit pour une entrée de type QAM-16

Nous retrouvons le même type de comportement que dans le cas des signaux à valeurs réelles. La même adéquation est aussi observée en comparant avec la méthode classique et celle proposée par [129]³, (voir le tableau 4.6). Il en est aussi de même pour le conditionnement numérique (tableau 4.7) dont l'amélioration est davantage marquée.

SNR (dB)	5	15	25	35
Méthode Classique (dB)	-6.199	-15.150	-25.019	-35.005
méthode de Cheng et Powers (dB)	-6.193	-15.018	-23.954	-29.233
Méthode Orthogonale (dB)	-6.185	-15.023	-23.959	-29.258

TAB. 4.6 – EQMN pour une entrée QAM-16 ($N = 16000$)

Entrée	Méthode classique	Méthode orthogonale
QAM-16	6.146×10^3	1.146
QAM-64	9.15×10^4	1.144

TAB. 4.7 – Conditionnement de la matrice des moments de l'entrée d'un modèle de Volterra cubique ; entrée de type QAM-16 et QAM-64

Remarque 4.5 *La méthode orthogonale que nous avons proposée se distingue de celles de Tseng et Powers ([129, 131]), tant dans le cas complexe que réel, par la simplicité des équations d'une part et surtout par le fait qu'elle améliore le conditionnement numérique de la matrice des moments de l'entrée. Ce point particulier motive le recours au formalisme des bases de polynômes orthogonaux pour accélérer la convergence des algorithmes adaptatifs.*

4.3 Structure orthogonale des filtres issus de la BFOR

Nous avons montré dans la section précédente que l'estimation paramétrique d'un modèle de Volterra peut-être améliorée en exploitant certaines propriétés de l'entrée du système. Tout particulièrement l'hypothèse *i.i.d.* a été posée. Nous n'avons pas considéré le cas éventuel où le modèle serait d'une complexité paramétrique importante et encore moins le cas d'une entrée colorée.

S'agissant de la complexité paramétrique, dans le chapitre 3 nous avons montré qu'un développement sur une base de fonctions orthogonales rationnelles (BFOR) permet d'obtenir une représentation davantage parcimonieuse. Nous allons montrer comment l'utilisation de ces bases permet d'améliorer le conditionnement de la matrice des moments de l'entrée et comment mettre en place une structure orthogonale pour les filtres issus de ces bases. Nous considérons d'abord le cas du noyau de Volterra du premier ordre, i.e. le cas linéaire avant de considérer le cas des noyaux d'ordre supérieur.

³Les principales équations de cette méthode sont rappelées dans l'annexe D.2

4.3.1 Cas linéaire

4.3.1.1 Orthogonaliseur en escalier

Considérons les entrées filtrées $s_k(n) = \sum_{i=0}^{\infty} b_k(i)u(n-i)$ issues d'une BFOR $\mathfrak{B}$ et la transformation linéaire caractérisée par la matrice $\mathbf{W}$ qui permette leur orthogonalisation. Posons :

$$\begin{aligned} S(n) &= (s_0(n) \ s_1(n) \cdots s_{K-1}(n))^T \\ \check{S}(n) &= (\check{s}_0(n) \ \check{s}_1(n) \cdots \check{s}_{K-1}(n))^T \end{aligned}$$

$S(n)$ et $\check{S}(n)$ sont respectivement les vecteurs de signaux d'entrée filtrée avant et après le module d'orthogonalisation.

$$\check{S}(n) = \mathbf{W}S(n) \quad (4.62)$$

et

$$\mathbf{\Gamma}_{\check{S}} = \mathbb{E}[\check{S}(n)\check{S}^T(n)] = \mathbf{W}\mathbf{\Gamma}_S\mathbf{W}^T$$

où $\mathbf{\Gamma}_{\check{S}}$ est une matrice diagonale et $\mathbf{W}$ une matrice triangulaire inférieure dont les éléments diagonaux sont égaux à un.

Pour déterminer la matrice d'orthogonalisation $\mathbf{W}$, nous allons utiliser une structure de filtre dite filtre en escalier. Cette structure a été introduite par Ahmed et Youn [2] pour la prédiction linéaire adaptative. Dans un premier temps, dans un contexte non adaptatif, nous exploitons cette structure pour orthogonaliser les entrées filtrées par la BFOR.

La matrice $\mathbf{W}$ peut s'écrire comme suit :

$$\mathbf{W} = \begin{pmatrix} 1 & & & & & \\ w_{1,1} & 1 & & & & \\ w_{2,2} & w_{2,1} & & & & \\ \vdots & \vdots & \ddots & & & \\ w_{K-2,K-2} & w_{K-2,K-3} & \cdots & 1 & 0 & \\ w_{K-1,K-1} & w_{K-1,K-2} & \cdots & w_{K-1,1} & 1 & \end{pmatrix}$$

En posant $w_{i,j} = -v_{i,j}$ on obtient les K équations suivantes :

$$\begin{aligned} \check{s}_0(n) &= s_0(n) \\ \check{s}_k(n) &= s_k(n) - \sum_{j=1}^k v_{k,j}s_{k-j}(n), \quad k = 1, 2, \dots, K-1 \end{aligned}$$

$\check{s}_k(n)$ peut alors être interprété comme étant l'erreur de prédiction de l'entrée filtrée $s_k(n)$ à partir des entrées filtrées issues de toutes les fonctions de base précédentes.

La matrice d'orthogonalisation $\mathbf{W}$ peut être factorisée en $K-1$ matrices $\mathbf{W}_k$, $k = 1, 2, \dots, K-1$ telle que :

$$\mathbf{W} = \mathbf{W}_{K-1}\mathbf{W}_{K-2}\cdots\mathbf{W}_k\cdots\mathbf{W}_1 \quad (4.63)$$

où

$$\mathbf{W}_k = \begin{pmatrix} 1 & & & & & & \\ 0 & 1 & & & & & \\ 0 & 0 & 1 & & & & \\ \vdots & \vdots & \cdots & 1 & & & \\ 0 & \cdots & 0 & \alpha_k^k & 1 & \cdots & \\ \vdots & & & \vdots & & \cdots & \vdots \\ 0 & \cdots & 0 & \alpha_{K-1}^k & \cdots & 0 & 1 \end{pmatrix} \quad (4.64)$$

A partir de cette factorisation, on constate que l'orthogonalisation peut être effectuée en $K - 1$ étapes. Pour illustrer la procédure, prenons le cas où $K = 4$. La factorisation de la matrice d'orthogonalisation est :

$$\mathbf{W} = \mathbf{W}_3 \mathbf{W}_2 \mathbf{W}_1$$

avec :

$$\mathbf{W}_1 = \begin{pmatrix} 1 & & & \\ \alpha_1^1 & 1 & & \\ \alpha_2^1 & 0 & 1 & \\ \alpha_3^1 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{W}_2 = \begin{pmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & \alpha_2^2 & 1 & \\ 0 & \alpha_3^2 & 0 & 1 \end{pmatrix} \quad \mathbf{W}_3 = \begin{pmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & 0 & 1 & \\ 0 & 0 & \alpha_3^3 & 1 \end{pmatrix}$$

Posons $y_{0,0}(n) = s_0(n)$ et rappelons que dans notre cas $S(n) = (s_0(n) \ s_1(n) \ s_2(n) \ s_3(n))^T$. Une première "marche" de l'escalier est obtenue en calculant $\mathbf{W}_1 S(n)$:

$$\begin{aligned} \mathbf{W}_1 S(n) &= \begin{pmatrix} 1 & & & \\ \alpha_1^1 & 1 & & \\ \alpha_2^1 & 0 & 1 & \\ \alpha_3^1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{0,0}(n) \\ s_1(n) \\ s_2(n) \\ s_3(n) \end{pmatrix} = \begin{pmatrix} y_{0,0}(n) \\ s_1(n) + \alpha_1^1 y_{0,0}(n) \\ s_2(n) + \alpha_2^1 y_{0,0}(n) \\ s_3(n) + \alpha_3^1 y_{0,0}(n) \end{pmatrix} \\ &= \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,1}(n) \\ y_{3,1}(n) \end{pmatrix} \end{aligned}$$

puis :

$$\begin{aligned} \mathbf{W}_2 \mathbf{W}_1 S(n) &= \begin{pmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & \alpha_2^2 & 1 & \\ 0 & \alpha_3^2 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,1}(n) \\ y_{3,1}(n) \end{pmatrix} = \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,1}(n) + \alpha_2^2 y_{1,1}(n) \\ y_{3,1}(n) + \alpha_3^2 y_{1,1}(n) \end{pmatrix} \\ &= \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,2}(n) \\ y_{3,2}(n) \end{pmatrix} \end{aligned}$$

et enfin

$$\begin{aligned} \mathbf{W}_3 \mathbf{W}_2 \mathbf{W}_1 S(n) &= \begin{pmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & 0 & 1 & \\ 0 & 0 & \alpha_3^3 & 1 \end{pmatrix} \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,2}(n) \\ y_{3,2}(n) \end{pmatrix} = \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,2}(n) \\ y_{3,2}(n) + \alpha_3^3 y_{2,2}(n) \end{pmatrix} \\ &= \begin{pmatrix} y_{0,0}(n) \\ y_{1,1}(n) \\ y_{2,2}(n) \\ y_{3,3}(n) \end{pmatrix} \end{aligned}$$

Nous allons montrer que les signaux $y_{i,i}(n)$, $i = 0, 1, 2, 3$ correspondent respectivement aux signaux $\check{s}_0(n)$, $\check{s}_1(n)$, $\check{s}_2(n)$ et $\check{s}_3(n)$. A chaque étape un seul signal $\check{s}_i(n)$ est déterminé. Les sorties à chaque étape peuvent être caractérisées comme suit :

– Etape 1 :

$y_{1,1}(n)$ est l'erreur de prédiction de $s_1(n)$ en utilisant $s_0(n)$

$y_{2,1}(n)$ est l'erreur de prédiction de $s_2(n)$ en utilisant $s_0(n)$

$y_{3,1}(n)$ est l'erreur de prédiction de $s_3(n)$ en utilisant $s_0(n)$

– Etape 2 :

$y_{2,2}(n)$ est l'erreur de prédiction de $s_2(n)$ en utilisant $s_0(n)$ et $s_1(n)$

$y_{3,2}(n)$ est l'erreur de prédiction de $s_3(n)$ en utilisant $s_0(n)$ et $s_1(n)$

– Etape 3 :

$y_{3,3}(n)$ est l'erreur de prédiction de $s_3(n)$ en utilisant $s_0(n)$, $s_1(n)$ et $s_2(n)$

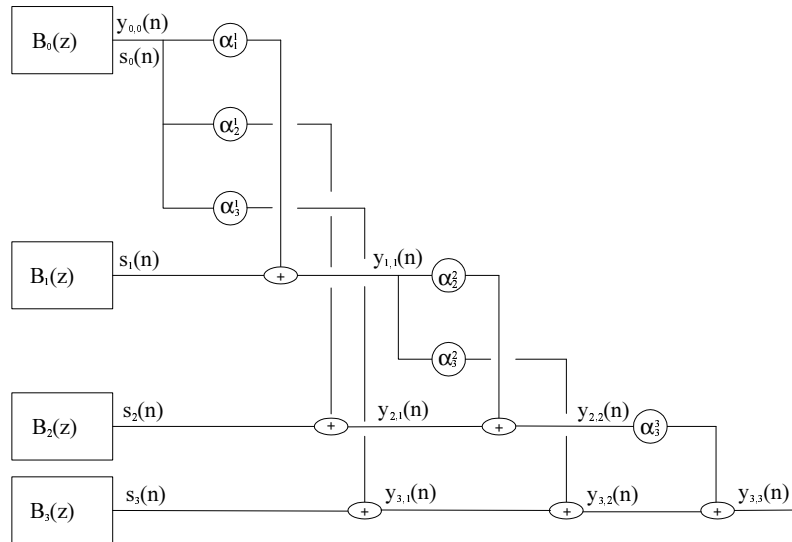


FIG. 4.6 – Structure en escalier pour l'orthogonalisation des entrées filtrées par la BFOR pour $K = 4$

D'une manière générale, pour réaliser la transformation (4.62) mettant en jeu un vecteur de régression S ayant K composantes, la structure en escalier comporte $K - 1$ étapes. L'étape i , $i = 1, 2, \dots, K - 1$, contient $K - i$ coefficients α_j^i , $j = i, i + 1, \dots, K - 1$, soit au total $K(K - 1)/2$ coefficients. Les signaux $y_{j,i}(n)$ en sortie de chaque étape i sont reliés à ceux de l'étape précédente par la relation suivante :

$$y_{j,i}(n) = y_{j,i-1}(n) + \alpha_j^i y_{i-1,i-1}(n), j \geq i \quad (4.65)$$

avec $y_{j,0}(n) = s_j(n)$, $j = 0, 1, \dots, K - 1$.

Le coefficient α_j^i est "local" à l'étage i et $y_{j,i}(n)$ est l'erreur de prédiction de $y_{j,i-1}(n)$ à partir de $y_{i-1,i-1}(n)$. Par ailleurs, en développant la relation ci-dessus, on montre que les signaux $y_{i,i}$ sont reliés par la relation suivante :

$$y_{i,i}(n) = y_{i,0}(n) + \sum_{j=1}^i \alpha_j^i y_{j-1,j-1}(n), \quad i = 1, 2, \dots, K - 1 \quad (4.66)$$

Le principe de transformation des signaux $s_i(n)$ en $\check{s}_i(n) = y_{i,i}(n)$ étant posé, comment choisir les coefficients α_j^i en vue de réaliser une orthogonalisation ?

Nous allons chercher les coefficients α_j^i tels que :

$$\mathbb{E} \{ y_{j,i}(n) y_{i-1,i-1}(n) \} = 0 \quad (4.67)$$

C'est le principe d'orthogonalité bien connu en filtrage optimal. Cela équivaut à minimiser l'erreur quadratique de prédiction et donc à minimiser la fonction de coût suivante :

$$J_{j,i} = \mathbb{E} \{ y_{j,i}^2(n) \}$$

Le calcul du gradient par rapport au coefficient α_j^i donne alors :

$$\begin{aligned} \frac{\partial J_{j,i}}{\partial \alpha_j^i} &= 2 \mathbb{E} \left\{ y_{j,i}(n) \frac{\partial y_{j,i}(n)}{\partial \alpha_j^i} \right\} \\ &= 2 \mathbb{E} \{ y_{j,i}(n) y_{i-1,i-1}(n) \} \\ &= 2 \mathbb{E} \{ y_{j,i-1}(n) y_{i-1,i-1}(n) \} + 2 \alpha_j^i \mathbb{E} \{ y_{i-1,i-1}^2(n) \} \end{aligned}$$

En annulant le gradient de la fonction de coût considérée, nous obtenons les coefficients optimaux de l'escalier :

$$\alpha_j^i = - \frac{\mathbb{E} \{ y_{j,i-1}(n) y_{i-1,i-1}(n) \}}{\mathbb{E} \{ y_{i-1,i-1}^2(n) \}}, \quad i = 1, 2, \dots, K - 1, j \geq i \quad (4.68)$$

Le coefficient optimal α_j^i rend le signal $y_{j,i}$, avec $j \geq i$, orthogonal au signal $y_{i-1,i-1}$ (principe d'orthogonalité (4.67)). En particulier, $y_{i,i}$ est orthogonal à $y_{i-1,i-1}$. De manière itérative on peut vérifier que les signaux $y_{i,i}$, $i = 0, \dots, K - 1$, sont deux à deux

orthogonaux. En effet : comme nous l'avons stipulé ci-dessus $y_{0,0}$ et $y_{1,1}$ sont orthogonaux d'une part et $y_{1,1}$ et $y_{2,2}$ le sont de l'autre. Sachant que $y_{2,2}$ est la combinaison linéaire de $y_{1,1}$ et de $y_{2,1}$ et que $y_{2,1}$ est orthogonal à $y_{0,0}$ (condition d'orthogonalisation (4.67)), on conclut que $y_{2,2}$ est aussi orthogonal à $y_{0,0}$. En itérant ce processus, on vérifie l'orthogonalité des $y_{i,i}$.

Nous allons montrer, à présent, que cet algorithme réalise une orthogonalisation de Gram-Schmidt. A partir des relations, (4.66) et (4.68), on obtient

$$y_{i,i}(n) = y_{i,0}(n) - \sum_{j=1}^i \frac{\mathbb{E}[y_{i,j-1}(n)y_{j-1,j-1}(n)]}{\mathbb{E}[y_{j-1,j-1}^2(n)]} y_{j-1,j-1}(n)$$

On a par ailleurs :

$$\begin{aligned} \mathbb{E}[y_{i,j-1}(n)y_{j-1,j-1}(n)] &= \mathbb{E}\left[\left(y_{i,j-2}(n) + \alpha_i^{j-1}y_{j-2,j-2}(n)\right)y_{j-1,j-1}(n)\right] \\ &= \mathbb{E}[y_{i,j-2}(n)y_{j-1,j-1}(n)] \\ &\quad + \alpha_i^{j-1}\mathbb{E}[y_{j-2,j-2}(n)y_{j-1,j-1}(n)] \\ &= \mathbb{E}[y_{i,j-2}(n)y_{j-1,j-1}(n)] \end{aligned}$$

De proche en proche, on montre que

$$\mathbb{E}[y_{i,j-1}(n)y_{j-1,j-1}(n)] = \mathbb{E}[y_{i,0}(n)y_{j-1,j-1}(n)]$$

D'où :

$$y_{i,i}(n) = y_{i,0}(n) - \sum_{j=1}^i \frac{\mathbb{E}[y_{i,0}(n)y_{j-1,j-1}(n)]}{\mathbb{E}[y_{j-1,j-1}^2(n)]} y_{j-1,j-1}(n) \quad (4.69)$$

On retrouve bien la relation d'orthogonalisation de Gram-Schmidt.

Il est à signaler que le calcul des coefficients optimaux (4.68) nécessite l'évaluation d'une moyenne statistique. En pratique, sous l'hypothèse d'ergodicité, ces coefficients sont donnés par :

$$\hat{\alpha}_j^i = - \frac{\sum_{n=0}^{N-1} y_{j,i-1}(n)y_{i-1,i-1}(n)}{\sum_{n=0}^{N-1} y_{i-1,i-1}^2(n)} \quad (4.70)$$

En résumé, la matrice $\mathbf{W}$, factorisée selon (4.63), réalise une orthogonalisation des signaux s_i lorsque les coefficients α_j^i de ses différents facteurs sont donnés par (4.68). En considérant que nous avons à disposition les entrées filtrées $s_k(n)$, $k = 0, \dots, K-1$, $n = 0, \dots, N-1$, l'algorithme d'orthogonalisation en bloc que nous proposons est le suivant :

Algorithme 4.1 Orthogonalisation en bloc des entrées filtrées par la BFOR

1. Initialisation :
 - $y_{i,0}(n) = s_i(n)$, $i = 0, \dots, K-1$, $n = 0, \dots, N-1$
 - $\check{s}_0(n) = y_{0,0}(n)$, $n = 0, \dots, N-1$
 - $i=1$
2. (a) Calculer α_j^i , $j = i, \dots, K-1$, selon (4.70);
 - (b) Calculer $y_{j,i}(n)$, $j = i, \dots, K-1$, $n = 0, \dots, N-1$, selon (4.65);
 - (c) Prendre $\check{s}_i(n) = y_{i,i}(n)$, $n = 0, \dots, N-1$
 - (d) Incrémenter i puis retourner à (2a) tant que $i \leq K-1$ sinon arrêter.

Une autre manière de construire la matrice d'orthogonalisation W serait de réaliser la décomposition de Cholesky de la matrice d'autocorrélation $\mathbb{E}\{SS^T\}$ puis de prendre W comme étant l'inverse du facteur de Cholesky. Cependant, la décomposition de Cholesky a un coût de calcul en $O(K^3)$ (la matrice à décomposer étant d'ordre K)[48]. Il existe des versions rapides, i.e. ayant un coût calculatoire moins élevé, mais elles nécessitent que la matrice à décomposer ait une structure de Toeplitz. Pour notre application, la structure en escalier que nous avons présentée, a donc de réels atouts car son coût calculatoire est en $O(K^2)$. De plus, elle peut être utilisée dans un contexte adaptatif via la version adaptative que nous allons présenter dans le paragraphe suivant.

Pour trouver la relation entre les coefficients de Fourier de la structure transversale et ceux de la structure en escalier, on rappelle que la sortie instantanée du modèle s'écrit d'une part :

$$y(n) = G^T S(n)$$

et de l'autre

$$y(n) = \check{G}^T \check{S}(n) = \check{G}^T \mathbf{W} S(n)$$

d'où :

$$G = \mathbf{W}^T \check{G} \quad (4.71)$$

Dans les deux propositions ci-dessous, nous montrons l'apport des BFOR quant au conditionnement numérique de la matrice d'autocorrélation.

Proposition 4.3 *Le filtrage par les filtres de la BFOR préserve la blancheur d'une séquence d'entrée ; i.e. les entrées filtrées sont aussi non corrélées.*

Preuve : Considérons les filtres de la BFOR suivants $B_k(z)$, $k = 0, 1, \dots$. Les séquences temporelles correspondant aux réponses impulsionnelles des filtres de la BFOR peuvent être rangées dans la matrice ci-dessous :

$$\mathbf{B} = \begin{pmatrix} b_0(0) & b_0(1) & \dots \\ b_1(0) & b_1(1) & \dots \\ \vdots & \vdots & \dots \\ b_{K-1}(0) & b_{K-1}(1) & \dots \end{pmatrix}$$

L'orthonormalité de la base permet d'écrire que $\mathbf{B}\mathbf{B}^T = \mathbf{I}_K$. Considérons à présent le filtrage d'un signal blanc de variance σ_u^2 . Le vecteur des entrées filtrées s'écrit :

$$S(n) = \mathbf{B}U(n)$$

où

$$U(n) = (u(n) \ u(n-1) \ \dots)^T$$

La matrice d'auto-corrélation du vecteur des entrées filtrées est alors :

$$\begin{aligned} \Gamma_S &= \mathbb{E}[S(n)S^T(n)] = \mathbf{B}\mathbb{E}[U(n)U^T(n)]\mathbf{B}^T \\ &= \sigma_u^2 \mathbf{B}\mathbf{B}^T \\ &= \sigma_u^2 \mathbf{I}_K \end{aligned}$$

Les entrées filtrées sont donc non corrélées. ■

Proposition 4.4 *La matrice d'autocorrélation des entrées filtrées par la BFOR a un nombre condition inférieur ou égal à celui de la matrice d'autocorrélation de la séquence d'entrée.*

Preuve : Comme indiqué précédemment, la matrice de corrélation des entrées filtrées s'écrit : $\Gamma_S = \mathbf{B}\Gamma_U\mathbf{B}^T$. Selon [48], $\chi(\Gamma_S) \leq \chi(\mathbf{B})\chi(\Gamma_U)\chi(\mathbf{B}^T)$. Sachant que $\mathbf{B}\mathbf{B}^T = \mathbf{I}$ alors $\chi(\mathbf{B})\chi(\mathbf{B}^T) = 1$. D'où : $\chi(\Gamma_S) \leq \chi(\Gamma_U)$. ■

Notons qu'en pratique la réponse impulsionnelle des filtres issus de la BFOR est tronquée à un ordre fini, ce qui peut limiter l'exploitation des propriétés énoncées dans les propositions 4.3 et 4.4.

4.3.1.2 Filtre adaptatif en escalier

En traitement du signal, un bon nombre d'applications nécessite une structure de filtre adaptatif. Lorsque le système est modélisé à l'aide des BFOR, l'aspect adaptatif ne concerne que les coefficients de Fourier si on considère que la base est fixée. Le vecteur des paramètres à adapter est celui des coefficients de Fourier tandis que le vecteur d'observation est celui des entrées filtrées. Pour signifier le caractère adaptatif du filtre, on note :

$$y(n) = G^T(n)S(n) \tag{4.72}$$

Pour une structure transversale de la BFOR, l'adaptation des coefficients de Fourier peut se faire à l'aide des algorithmes classiques tels que *LMS*, *NLMS* ou *RLS*. Pour le cas des structures orthogonales, [42] ont proposé un algorithme de type gradient normalisé pour le filtre de Laguerre en treillis. Dans ce qui suit, nous présentons un algorithme adaptatif pour la structure orthogonale des BFOR que nous avons proposée dans la section précédente.

Le problème se résume à la construction adaptative de la matrice d'orthogonalisation $\mathbf{W}$. L'algorithme adaptatif inhérent à cette structure proposée par [2] a été par la suite modifié pour améliorer les performances des filtres adaptés avec l'algorithme *LMS* dans le domaine de la transformée en cosinus discrète (DCT, *Discrete Cosinus Transform*) pour des systèmes linéaires [106] et dans le domaine de la transformée en ondelettes discrètes (DWT, *Discrete Wavelet transform*) pour le modèle de Volterra [60]. Nous réalisons une extension de cet algorithme dans le cas de l'orthogonalisation des entrées filtrées par la BFOR de Ninness et Gustafsson.

Dans un contexte adaptatif, l'équation (4.65) s'écrit :

$$y_{j,i}(n) = y_{j,i-1}(n) + \alpha_j^i(n)y_{i-1,i-1}(n) \quad (4.73)$$

Contrairement à ce que nous avons proposé dans la section précédente, nous prendrons $J_{j,i} = y_{j,i}^2(n)$. Les coefficients optimaux de l'escalier sont alors adaptés en minimisant la valeur de cette fonction de coût. L'algorithme du gradient nous donne :

$$\alpha_j^i(n+1) = \alpha_j^i(n) - \frac{1}{2}\mu \frac{\partial y_{j,i}^2(n)}{\partial \alpha_j^i(n)}$$

d'où :

$$\alpha_j^i(n+1) = \alpha_j^i(n) - \mu y_{j,i}(n)y_{i-1,i-1}(n) \quad (4.74)$$

Une version normalisée de cet algorithme est obtenue en normalisant la constante μ par la variance de l'erreur de prédiction $y_{j,i}$. En nommant σ_j^i ladite variance on a alors :

$$\alpha_j^i(n+1) = \alpha_j^i(n) - \frac{\mu}{\sigma_j^i(n)} y_{j,i}(n)y_{i-1,i-1}(n) \quad (4.75)$$

Cette variance est estimée suivant la relation :

$$\sigma_j^i(n+1) = \beta \sigma_j^i(n) + (1-\beta)y_{j,i}^2(n); \quad 0 < \beta < 1 \quad (4.76)$$

Algorithme 4.2 Orthogonalisation adaptative des entrées filtrées par la BFOR

1. Initialisation :

- Pour $i = 1, \dots, K-1$, $j = i, \dots, K-1$, prendre les $\alpha_j^i(0)$ nuls et les $\sigma_j^i(0)$ égaux à une petite valeur positive ;
- Choisir β proche de 1 et un pas d'adaptation μ ;
- $y_{j,i}(0) = y_{j,i-1}(0)$
- $\check{s}_i(0) = y_{i,i}(0)$, $i = 0, \dots, K-1$

2. (a) Incrémenter n ;

(b) $i = 1$;

(c) Adapter $\sigma_j^i(n)$ selon (4.76) ;

(d) Calculer $y_{j,i}(n)$, $j = i, \dots, K-1$, selon (4.73) ;

4.3. Structure orthogonale des filtres issus de la BFOR

- (e) Prendre $\check{s}_i(n) = y_{i,i}(n)$;
- (f) Adapter $\alpha_j^i(n)$, $j = i, \dots, K-1$, selon (4.75) ;
- (g) Incrémenter i puis retourner à (2c) tant que $i \leq K-1$ sinon retourner à (2a) ;

Cet algorithme a une complexité de calcul en $O(K^2)$ (voir Tab. 4.8) :

Additions	$K(K-1)/2$
Multiplications	$3K(K+1)/2$
Divisions	$K(K+1)$

TAB. 4.8 – Nombre d'opérations élémentaires à chaque itération

Une mesure de l'efficacité de l'orthogonalisation est donnée par :

$$\eta = 1 - \frac{\sum_{i=1}^{K-2} \sum_{j=i+1}^{K-1} \Gamma_{\check{s}}^2(i, j)}{\sum_{i=1}^{K-2} \sum_{j=i+1}^{K-1} \Gamma_{\check{s}}^2(i, j)} \quad (4.77)$$

Dans l'exemple suivant, nous allons associer à la structure orthogonale en escalier l'algorithme *LMSN* (LMS-Newton) pour l'adaptation des coefficients de Fourier, dans le contexte de l'identification des systèmes. Les coefficients de Fourier sont donnés par :

$$\check{G}(n+1) = \check{G}(n) + \mu \Gamma_{\check{s}}^{-1}(n) \check{S}(n) e(n) \quad (4.78)$$

où $e(n)$ est l'erreur instantanée entre la sortie réelle du système à identifier et la sortie du modèle. En approchant la matrice d'autocorrélation $\Gamma_{\check{s}}(n)$ par une matrice diagonale, on obtient alors :

$$\check{g}_k(n+1) = \check{g}_k(n) + \frac{\mu}{\sigma_k^2(n)} \check{s}_k(n) e(n), \quad k = 0, 1, \dots, K-1 \quad (4.79)$$

où $\sigma_k^2(n)$ est la variance de $\check{s}_k(n)$ estimée comme suit :

$$\sigma_k^2(n+1) = \beta \sigma_k^2(n) + (1-\beta) \check{s}_k^2(n), \quad k = 0, 1, \dots, K-1 \quad 0 < \beta < 1 \quad (4.80)$$

4.3.2 Filtre de Volterra-BFOR adaptatif basé sur l'orthogonaliseur en escalier

En considérant que la BFOR a été bien choisie, le problème de l'estimation des paramètres du filtre de Volterra-BFOR se réduit à l'estimation des coefficients de Fourier. Pour ce faire, nous pouvons recourir aux algorithmes de type LMS relativement simples

à implémenter et ayant un coût de calcul raisonnable. Cependant les caractéristiques de convergence de ce type d'algorithmes dépendent fortement de la corrélation de la séquence d'entrée. La convergence est d'autant plus rapide que le rapport entre la plus grande et la plus faible valeur propre de la matrice d'autocorrélation de l'entrée est proche de 1. Dans le filtrage non-linéaire, même si le signal d'entrée est blanc, la présence des termes non-linéaires provoque un mauvais conditionnement de la matrice des moments de l'entrée. Il en résulte une convergence très lente des algorithmes de type LMS appliqués aux filtres de Volterra [86]. En recourant à l'algorithme RLS, la convergence est nettement améliorée mais au prix d'une complexité de calcul élevée. Pour un filtre de Volterra d'ordre P et de mémoire M , la complexité de calcul de l'algorithme RLS est de $O(M^{2P})$ opérations alors que l'algorithme LMS n'en requiert que $O(M^P)$. Lorsque les noyaux de Volterra sont développés sur une BFOR d'ordre K , les complexités respectives des algorithmes LMS et RLS sont en $O(K^P)$ et $O(K^{2P})$. Une manière d'améliorer la rapidité de convergence de l'algorithme LMS est donc de considérer des structures orthogonales [60].

La mise en commun du développement des noyaux de Volterra sur une BFOR avec le module d'orthogonalisation de Gram-Schmidt de type escalier est illustré par la figure 4.7. Nous recourons au même module d'orthogonalisation que dans le cas linéaire. La

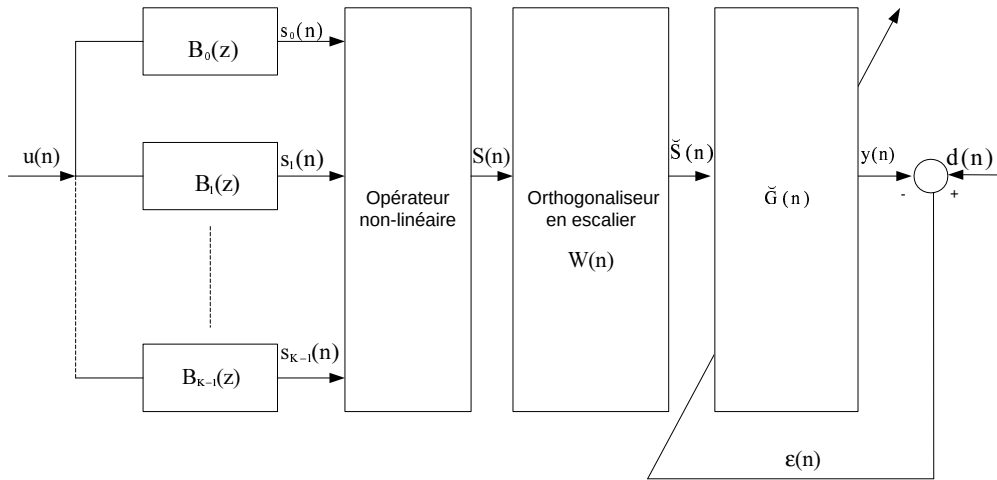


FIG. 4.7 – Filtrage adaptatif de Volterra avec structure orthogonale

seule différence est que le vecteur d'observation $S(n)$ contient désormais les entrées filtrées et leurs produits croisés. Par suite nous obtenons l'algorithme LMSN suivant :

$$\check{G}(n+1) = \check{G}(n) + \mu \Gamma_{\check{S}}^{-1}(n) \check{S}(n) \varepsilon(n)$$

où $\varepsilon(n)$ est l'erreur instantanée entre la sortie désirée et la sortie du filtre, et $\Gamma_{\check{S}}$ une matrice supposée diagonale. La complexité de l'algorithme, essentiellement due à l'adaptation de l'orthogonaliseur, est de $O(K^{2P})$, i.e. comparable à celui de l'algorithme RLS. Toutefois, lorsque les coefficients de l'orthogonaliseur deviennent relativement

constants le processus d'adaptation de l'orthogonaliseur est stoppé. La complexité de calcul est alors ramenée à celle d'un algorithme NLMS.

4.3.3 Simulations

4.3.3.1 Cas linéaire

Evaluation de l'orthogonalisation : Nous considérons un système de fonction de transfert

$$H(z) = \frac{z^2 + z}{z^3 - 0.28z^2 - 0.402z + 0.0756} \quad (4.81)$$

développé sur une BFOR de trois fonctions construites avec les pôles 0.77, -0.51 et 0.84 déterminés à l'aide de la procédure de recherche exhaustive décrite dans le chapitre 3. Le signal d'entrée est gaussien coloré.

En appliquant l'algorithme 4.1, l'orthogonaliseur optimal obtenu est :

$$\mathbf{W}_{opt} = \begin{pmatrix} 1 & 0 & 0 \\ -0.201 & 1 & 0 \\ 0.4454 & -4.7805 & 1 \end{pmatrix}$$

Les matrices de corrélation avant et après orthogonalisation, pour un traitement de $N = 2000$ données, sont :

$$\mathbf{\Gamma}_S = \begin{pmatrix} 0.756 & 0.152 & 0.391 \\ 0.152 & 0.046 & 0.151 \\ 0.391 & 0.151 & 0.862 \end{pmatrix}$$

$$\mathbf{\Gamma}_{\check{S},loc} = \begin{pmatrix} 0.756 & 3.79 \times 10^{-16} & -8.32 \times 10^{-18} \\ 3.79 \times 10^{-16} & 0.015 & -2.05 \times 10^{-15} \\ -8.32 \times 10^{-18} & -2.05 \times 10^{-15} & 0.316 \end{pmatrix}$$

où $\mathbf{\Gamma}_{\check{S},loc} = \frac{1}{N} \sum_{n=0}^{N-1} \check{S}(n)\check{S}^T(n)$ avec $\check{S}(n) = \mathbf{W}_{opt}S(n)$ et $\mathbf{\Gamma}_S = \frac{1}{N} \sum_{n=0}^{N-1} S(n)S^T(n)$.

Cet exemple illustre bien l'orthogonalisation ainsi réalisée. Il est à noter que cette orthogonalisation a été réalisée en utilisant l'ensemble de données disponibles. Faisons varier à présent le nombre de données utilisées pour la détermination de l'orthogonaliseur puis évaluons l'efficacité d'orthogonalisation (4.77) sur l'ensemble des 2000 données. La figure 4.8 donne la variation de l'efficacité d'orthogonalisation en fonction du nombre de données utilisées pour la détermination de l'orthogonaliseur. On constate qu'avec 200 données on obtient une efficacité d'orthogonalisation supérieure à 90% et qu'au delà de 500 données elle devient supérieure à 95%.

Par ailleurs, en utilisant l'algorithme adaptatif 4.2, la matrice de corrélation finale est :

$$\mathbf{\Gamma}_{\check{S},adapt} = \begin{pmatrix} 0.756 & 0.004 & -0.044 \\ 0.004 & 0.015 & -0.003 \\ -0.044 & -0.003 & 0.319 \end{pmatrix}$$

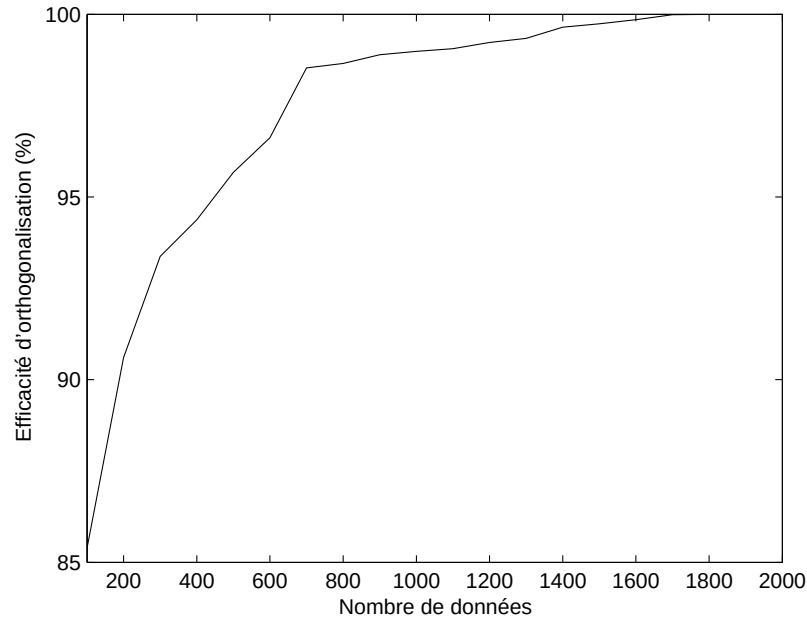


FIG. 4.8 – Efficacité d'orthogonalisation

On constate que les éléments de la diagonale principale sont beaucoup plus importants que tous les autres. La matrice tend à être diagonale comme on pouvait l'escompter. L'efficacité de l'orthogonalisation, selon (4.77) est de 99.00%. Cette matrice se rapproche de la matrice $\mathbf{\Gamma}_{\check{s},bloc}$ obtenue avec une orthogonalisation en bloc. L'orthogonaliseur obtenu est défini par :

$$\mathbf{W}_{adapt} = \begin{pmatrix} 1 & 0 & 0 \\ -0.196 & 1 & 0 \\ 0.419 & -4.940 & 1 \end{pmatrix}$$

Le pas d'adaptation de l'orthogonaliseur est égal à $\mu = 0.007$. A la convergence, l'orthogonaliseur obtenu est très proche de l'orthogonaliseur en bloc $\mathbf{W}_{opt}$. L'orthogonalisation adaptative tend donc vers l'orthogonalisation en bloc. Sur la figure 4.9, nous observons la convergence des coefficients $W_{2,1}$, $W_{3,1}$ et $W_{3,2}$.

Filtrage adaptatif avec module d'orthogonalisation : Considérons à présent l'identification du système (4.81) en considérant trois algorithmes adaptatifs différents. La figure 4.10 donne les performances des algorithmes NLMS, LMSN et RLS appliqués à la représentation du système sur une BFOR telle que décrite au paragraphe précédent. L'entrée est un signal gaussien coloré. Les pas d'adaptation des algorithmes ont été choisis de sorte que les performances en régime permanent soient similaires. Ils sont tels que $\mu_{NLMS-BFOR} = \mu_{NLMS-RIF} = 1.0$ et $\mu_{LMSN-BFOR} = 0.01$

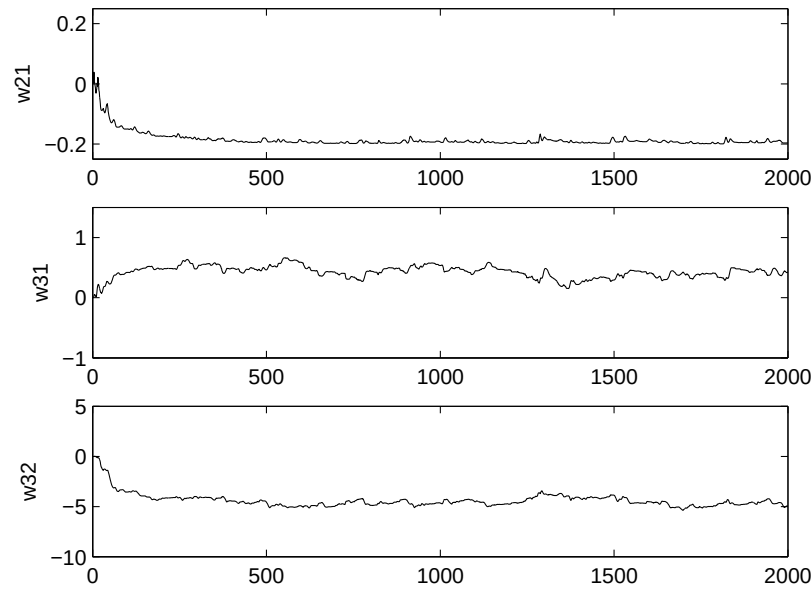


FIG. 4.9 – Variation des coefficients de l'orthogonaliseur

Dans le tableau 4.9, nous pouvons vérifier les améliorations successives apportées au conditionnement de la matrice d'autocorrélation par le pré-filtrage par la BFOR et par la structure en escalier.

Matrice d'autocorrélation	Γ_U	Γ_S	Γ_{ξ}
Nombre condition	4336	318.86	2.111

TAB. 4.9 – Nombre condition de la matrice d'autocorrélation de l'entrée

Il est à noter que pour stopper l'adaptation de la matrice d'orthogonalisation, le pas d'adaptation a été choisi tel que $\mu = 0.007$ pour les 200 premières itérations puis $\mu = 0$ pour le reste. Outre le fait de stabiliser la matrice d'orthogonalisation, ce choix permet de réduire la charge de calcul de l'algorithme. En effet, une fois la matrice d'orthogonalisation fixée, l'orthogonalisation ne requiert plus que K multiplications et K additions supplémentaires.

Comme on peut le constater sur la figure 4.10, l'algorithme *LMSN* converge plus vite que l'algorithme *NLMS* et moins vite que *RLS*. Cependant une fois la matrice d'orthogonalisation fixée, les performances sont similaires à celles de *RLS* avec une complexité de calcul plus faible.

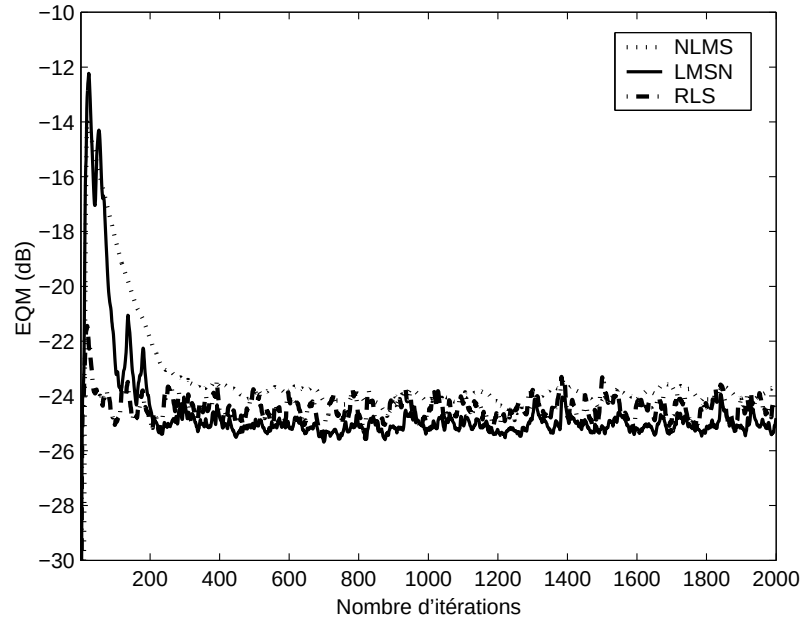


FIG. 4.10 – Comparaison de trois algorithmes adaptatifs pour une entrée gaussienne colorée (BFOR tronquée à $K = 3$)

4.3.3.2 Cas non-linéaire

Considérons l'identification adaptative d'un système non linéaire constitué par la mise en cascade d'un filtre linéaire $H(z)$ et d'un polynôme $P(u)$.

$$H(z) = \frac{z+1}{z^3 + 1.12z^2 + 0.186z - 0.0756}, \quad P(u) = 0.99u - 0.35u^3$$

Le signal d'entrée est blanc de type PAM-4. Le bruit additif de sortie est gaussien et blanc (SNR= 30 dB). La BFOR utilisée est tronquée à $K = 3$ fonctions, ses pôles, $-0.74, -0.62$ et 0.2 , sont proches de ceux du filtre linéaire. L'orthogonaliseur est adapté durant les 200 premières itérations avec un pas d'adaptation $\mu = 0.04$. La figure 4.11 est obtenue en moyennant 100 expériences indépendantes. Nous retrouvons le même type de comportement que dans le cas linéaire traité dans la section précédente à savoir : l'algorithme LMSN converge plus vite que le NLMS mais moins vite que le RLS. Sa complexité de calcul globale est moins élevée que celle du RLS.

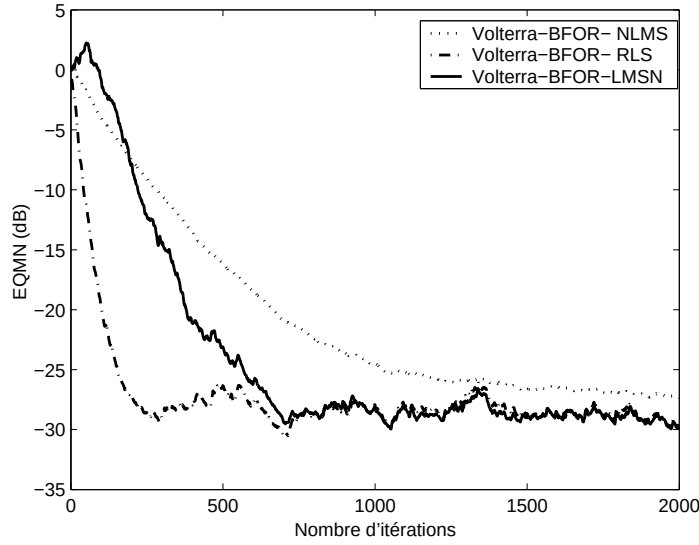


FIG. 4.11 – Comparaison de trois algorithmes d'adaptation d'un filtre de Volterra

4.4 Estimation découplée pour une entrée gaussienne colorée

Supposons que l'entrée est colorée, suit une distribution gaussienne et qu'en vue de réduire la complexité paramétrique du modèle de Volterra nous avons recours à un développement sur BFOR. Comme indiqué dans la section précédente, nous pouvons orthogonaliser les entrées filtrées par la BFOR en utilisant une structure de filtre en escalier. En sortie de ce module d'orthogonalisation nous obtenons une séquence non corrélée. Etant donné que le filtrage est linéaire et que la séquence d'entrée est gaussienne, nous obtenons donc en sortie du module d'orthogonalisation une séquence gaussienne *i.i.d* pour laquelle nous pouvons appliquer le formalisme des bases de polynômes orthogonaux. Nous avons noté qu'en suivant le même formalisme de construction que dans la section 4.2, dans le cas d'une séquence gaussienne, une base de polynômes d'Hermite est générée. Le principe de la méthode est décrit par la figure 4.12.

Considérons le développement d'un modèle de Volterra d'ordre P sur une BFOR à K fonctions. La relation d'entrée/sortie du filtre de Volterra-BFOR correspondant est :

$$y(n) = \sum_{p=1}^P \sum_{k_1=0}^{K-1} \cdots \sum_{k_p}^{K-1} g_{k_1, \dots, k_p} \prod_{j=1}^p s_{k_j}(n)$$

où $s_{k_j}(n) = \sum_{i=0}^p b_{k_j}(i)u(n-i)$. Posons $S_1(n) = (s_0(n) \cdots s_{K-1}(n))^T$.

Soit W la matrice d'orthogonalisation permettant de décorréler les composantes de $S_1(n)$. On a

$$\tilde{S}_1(n) = WS_1(n)$$

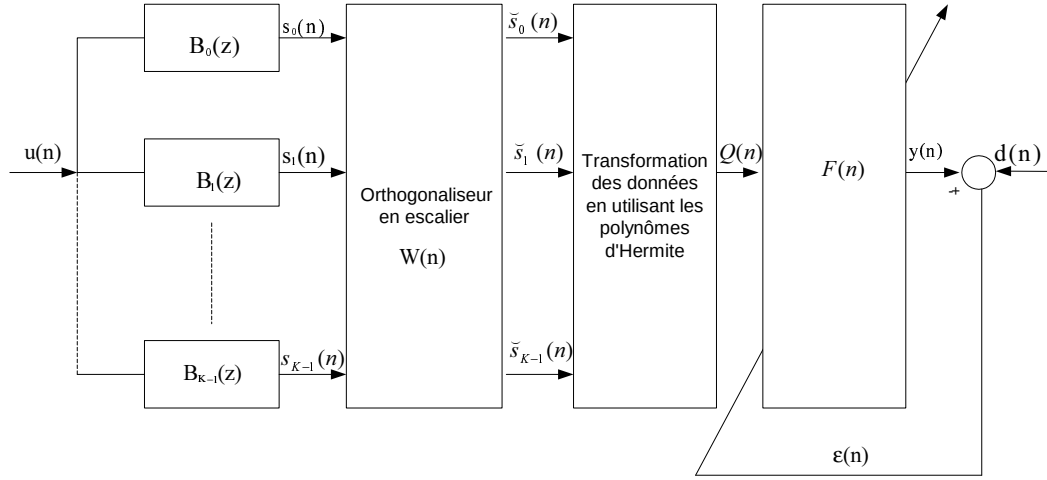


FIG. 4.12 – Estimation d'un modèle de Volterra à complexité réduite à l'aide des polynômes orthogonaux

En utilisant un orthogonaliseur en escalier, les composantes de $\check{S}_1(n)$ sont non corrélées et gaussiennes. La relation d'entrée-sortie du modèle de Volterra-BFOR devient alors :

$$y(n) = \sum_{p=1}^P \sum_{k_1=0}^{K-1} \cdots \sum_{k_p}^{K-1} \check{g}_{k_1, \dots, k_p} \prod_{j=1}^p \check{s}_{k_j}(n) = V(\check{S}(n))$$

$V(\check{S}(n))$ est un polynôme en les variables $\check{s}_k$, $k = 0, \dots, K-1$ qui peut être développé sur une base de polynômes orthogonaux multivariables construits à partir des polynômes d'Hermite monovariabiles $\mathbb{H}_p(\check{s}_k)$, où p est le degré du polynôme. Ceux-ci sont générés par la relation récurrente suivante [121] :

$$\mathbb{H}_{p+1}(\check{s}_k) = \frac{1}{\sqrt{p+1}} \left[\frac{\check{s}_k}{\sigma_{\check{s}_k}} \mathbb{H}_p(\check{s}_k) - \sqrt{p} \mathbb{H}_{p-1}(\check{s}_k) \right] \quad (4.82)$$

où $\mathbb{H}_0(\check{s}_k) = 1$ et $\sigma_{\check{s}_k}$ est la variance du signal gaussien $\check{s}_k$.

Le développement de $V(\check{S}(n))$ sur la base de polynômes orthogonaux multivariables $Q^{(p)}$ ⁴ est tel que :

$$y(n) = V(\check{S}(n)) = \sum_{p=1}^P \sum_{k_1=0}^{K-1} \cdots \sum_{k_p}^{K-1} f_{k_1, \dots, k_p} Q_{k_1, \dots, k_p}^{(p)}(n) = F^T Q(n)$$

L'estimation du modèle résultant équivaut donc à l'estimation des coefficients $f_{k_1, \dots, k_p}$ contenus dans le vecteur F . Dans le cas d'une estimation en bloc nous avons :

$$f_{k_1, \dots, k_p} = \mathbb{E} \left[d(n) Q_{k_1, \dots, k_p}^{(p)} \right]$$

⁴Le principe de construction de ces polynômes orthogonaux multivariables est le même que celui adopté pour les signaux PAM i.i.d.

où $d(n)$ est la sortie réelle du système.

L'avantage d'une telle structure d'estimation est de réduire le coût de calcul. Pour une estimation en bloc, le fait d'utiliser des polynômes orthogonaux permet d'éviter toute inversion matricielle ramenant ainsi le coût de calcul à $O(K^P)$ où P est l'ordre de non-linéarité du système et K le nombre de fonctions de base. En rappelant que l'orthogonalisation ne requiert que $O(K^2)$ opérations, nous obtenons une complexité de calcul global en $O(K^P)$. Dans la configuration décrite dans la section précédente, i.e. l'orthogonalisation est appliquée au vecteur des produits croisés des entrées filtrées, l'orthogonalisation requiert une complexité de calcul en $O(K^{2P})$. Nous pouvons donc noter une considérable réduction de la complexité de calcul. Dans le cas d'une estimation adaptative, avec la configuration de la figure 4.12, nous pouvons dériver un algorithme de type LMS de complexité de calcul en $O(K^P)$ et de convergence rapide.

Il est à noter que dans le cas d'une entrée gaussienne, à notre connaissance, la méthode proposée est la première qui couple la réduction de complexité à l'aide des BFOR, l'orthogonalisation en escalier et les bases de polynômes d'Hermite, soit trois niveaux d'orthogonalisation. Outre les travaux reportés dans [121], les polynômes d'Hermite ont aussi été appliqués pour l'estimation des modèles de Wiener-Hammerstein [10] et de Wiener-Laguerre [41].

Exemple 4.2 *En vue d'illustrer les potentialités de cette approche, nous considérons l'identification du système décrit dans la section précédente à l'aide d'une entrée gaussienne. Le signal d'entrée du système non-linéaire à identifier est un processus AR(1) obtenu en filtrant un signal blanc gaussien centré. Le filtre utilisé est défini par $H(z) = \frac{\sqrt{1-\alpha^2}}{1-\alpha z^{-1}}$. Il est à noter que le signal d'entrée est d'autant plus coloré que le pôle α du filtre AR est proche de 1. Pour différentes valeurs de ce pôle, nous avons évalué le nombre condition correspondant à chacune des configurations :*

- *Volterra* : le système est représenté par un modèle de Volterra de mémoire 15
- *Volterra+BFOR* : les noyaux de Volterra sont développés sur une BFOR à trois fonctions telles que décrites dans l'exemple de la section 4.3.3.2
- *Volterra+BFOR+Escalier* : Un module d'orthogonalisation est ajouté à la configuration précédente (voir la section 4.3.3.2)
- *Volterra+BFOR+Escalier+Polynômes Orthogonaux (PO)* : Le module d'orthogonalisation est ajouté immédiatement après le filtrage BFOR. Puis, des polynômes d'Hermite sont utilisés pour représenter la non-linéarité.

On peut noter que l'ajout des différentes structures d'orthogonalisation permet d'améliorer le conditionnement de la matrice des moments de l'entrée même dans le cas d'un signal blanc ($\alpha = 0$). Quelle que soit la configuration utilisée, le conditionnement de la dite matrice se détériore d'autant plus que le signal d'entrée est coloré. Cette détérioration est très peu marquée dans le cas de la quatrième configuration (Volterra+BFOR+Escalier+PO). Par ailleurs, il est nettement indiqué (figure 4.13) que la matrice des moments de l'entrée est mieux conditionnée dans cette dernière configuration. Théoriquement, on s'attend donc à avoir une nette amélioration de la vitesse de convergence de l'algorithme LMS lorsqu'une telle configuration est utilisée.

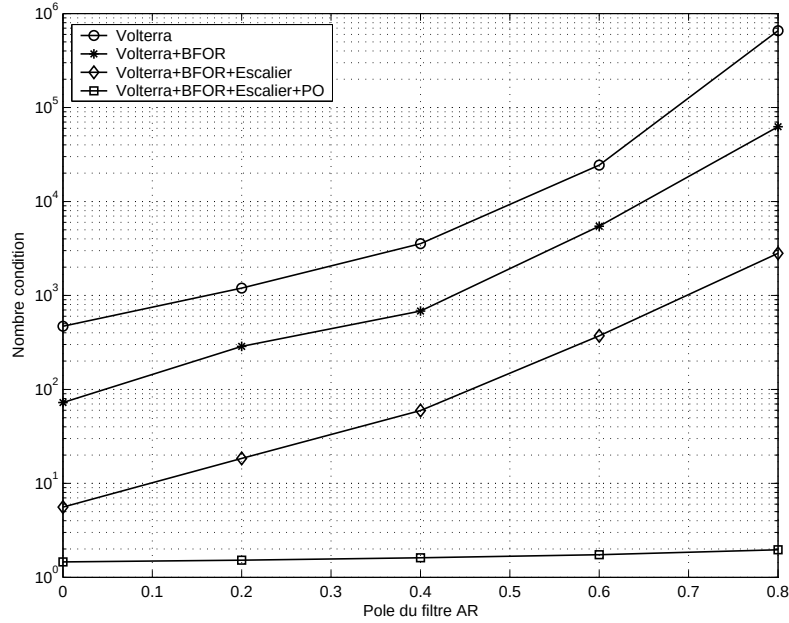


FIG. 4.13 – Nombre condition de la matrice des moments de l'entrée en fonction de la coloration du signal

4.5 Conclusion

Dans ce chapitre, nous avons présenté des méthodes orthogonales d'identification sous deux angles principaux.

Dans un premier temps nous avons exploité les propriétés statistiques des entrées de type PAM, QAM et PSK pour générer des bases de polynômes orthogonaux multivariés. En développant le modèle de Volterra sur une telle base, nous avons montré que l'estimation paramétrique du modèle, au sens EQMM, peut se faire sans aucune inversion matricielle. Nous avons aussi exploité le formalisme des bases de polynômes orthogonaux dans le cas de l'estimation adaptative. Il en résulte une amélioration de la vitesse de convergence de l'algorithme LMS sans un sur-coût de calcul.

Nous avons ensuite mis en place un module d'orthogonalisation des entrées filtrées par la BFOR. Nous avons ainsi couplé la notion de réduction de complexité du modèle et de simplification des procédures d'estimation.

Enfin, nous avons montré que dans le cas d'une entrée gaussienne le couplage du formalisme des polynômes orthogonaux et des bases de fonctions orthonormales rationnelles permet de simplifier davantage l'estimation paramétrique tant en bloc que de manière adaptative.

Chapitre 5

Contributions à l'identification et à l'égalisation aveugle et semi-aveugle

5.1 Introduction

Comme décrit dans le chapitre 1, la représentation de certains systèmes de communication est d'autant plus précise que le modèle utilisé est non-linéaire. Les modèles de Volterra et les réseaux de neurones sont les représentations non-linéaires les plus courantes. Les modèles de Volterra présentent l'avantage d'être linéaires par rapport aux paramètres et peuvent être vus comme une généralisation du modèle RIF. Cependant toute la difficulté dans l'utilisation de ce modèle vient de sa complexité paramétrique. En analysant davantage certains systèmes, on se rend compte qu'il est possible de les représenter par une cascade de sous-systèmes linéaires avec mémoire et non-linéaire sans mémoire. Ce type de représentation donne lieu, entre autres, au modèle de Wiener et à celui de Hammerstein. Ces modèles sont des cas particuliers de modèle de Volterra ayant une complexité paramétrique plus faible.

Dans ce chapitre, nous allons considérer le cas d'un système de communication sans fil pouvant être représenté par un modèle de Volterra. Selon que la liaison entre la station mobile et la station de base est considérée dans le sens ascendant (*uplink* en anglais), i.e. mobile vers station de base, ou dans le sens descendant (*downlink*), i.e. station de base vers mobile, le canal de communication peut-être vu respectivement comme un modèle de Wiener ou comme un modèle de Hammerstein. Certaines études s'intéressent à l'identification des sous-systèmes linéaire et non-linéaire. Dans notre étude nous considérons plutôt une modélisation globale en utilisant l'écriture en séries de Volterra des liaisons ascendante et descendante du canal.

L'identification et l'égalisation des modèles de Hammerstein et de Wiener ont générale-

ment été étudiées dans le cas supervisé. Il existe cependant quelques travaux consacrés à l'identification aveugle. Dans le cas du modèle de Hammerstein, seul le second ordre a été considéré en se basant sur l'exploitation des statistiques d'ordre supérieur, notamment les cumulants d'ordre trois, quatre et cinq de la sortie du système. Les méthodes proposées sont restreintes au cas des signaux gaussiens avec comme différence, une identification dans le domaine temporel [72] ou fréquentiel [76]. Une troisième méthode a été proposée par [109] dans le cadre d'une étude consacrée à l'identification aveugle des distorsions non-linéaires représentées par un modèle de Hammerstein. Là aussi, la gaussianité du signal est requise. Dans le cas du modèle de Wiener, un important travail a été présenté par [110]. Celui-ci ne considère que l'identification du sous-système linéaire pour le cas particulier des signaux d'entrées cyclostationnaires.

En ce qui concerne l'égalisation aveugle des canaux de communication non-linéaires, les canaux considérés dans la littérature sont de type *SIMO* ou *MIMO*. Cette structure de canal étant obtenue en utilisant un réseau d'antennes de réception ou en sur-échantillonnant la sortie du système. Giannakis et Serpedin [45] ont été les premiers à montrer que des systèmes linéaires de type FIR peuvent réaliser l'égalisation de type forçage à zéro (ZF) des canaux SIMO non-linéaires, sous certaines conditions. L'approche considérée est déterministe et relativement simple. Cependant elle présente un certain nombre de limitations. Tout d'abord elle requiert que la matrice associée au canal soit de rang plein et carrée. Ce qui n'est pas toujours possible. Ensuite, le noyau du premier ordre doit avoir la plus grande mémoire. Ces restrictions sur le rang de la matrice du canal et sur la mémoire du noyau du premier ordre ont été relaxées par Lopez-Valcarce et Dasgupta [78] dans une approche basée sur l'exploitation des statistiques du second-ordre, approche qui peut être considérée comme une extension au cas non-linéaire des travaux de Tong *et al.*[127]. Comme pour la plupart des algorithmes d'égalisation basés sur l'exploitation des statistiques du second ordre, cette approche requiert une complexité de calcul importante. Par ailleurs, il est à noter qu'une autre approche d'égalisation basée sur les statistiques du second ordre a été proposée par Raz et Van Veen [113], étendant au cas non-linéaire l'approche de Xu *et al.*[139]. Cependant, cette méthode requiert la linéarisabilité de chaque sous-canal par un filtre de Volterra d'ordre et de mémoire connus. Ce qui n'est pas toujours possible.

Dans notre étude, nous explorons une approche d'identification et égalisation semi-aveugles pour des canaux SISO et SIMO. Dans le cas SIMO, nous proposons aussi des procédures complètement aveugles. Dans un premier temps nous étudions la modélisation du système de communication que nous considérons avant de présenter chacune des nouvelles méthodes que nous proposons.

5.2 Accès à un réseau de communication sans fil via la fibre optique

Depuis quelques années, le monde des communications sans fil est entré dans une nouvelle phase. L'objet des transmissions migre de la voix vers les services multimédia, qui ont besoin de liaisons radio à large bande. C'est notamment l'enjeu principal de la téléphonie mobile dite de troisième génération. Par ailleurs, le nombre d'utilisateurs croît si vite qu'il est nécessaire d'augmenter les ressources physiques pour satisfaire les utilisateurs. Pour ce faire, le recourt à une architecture de réseau basée sur des micro et des pico cellules est une solution très attractive en ce qu'elle augmente la réutilisation des fréquences. Dans cette configuration, il est nécessaire d'augmenter les points d'accès au réseau. Pour des questions évidentes de coût, les stations de base conventionnelles (BST) sont remplacées par des points d'accès radio de puissance moindre (RAP : Radio Access Point). Ceux-ci devraient être robustes, de complexité faible et connectés à une BST via une fibre optique. Une telle architecture est représentée par la figure 5.1. Les étapes de traitement de données sont, quant à elles, décrites par les figures 5.2 et 5.3 respectivement pour la liaison ascendante et pour la liaison descendante.

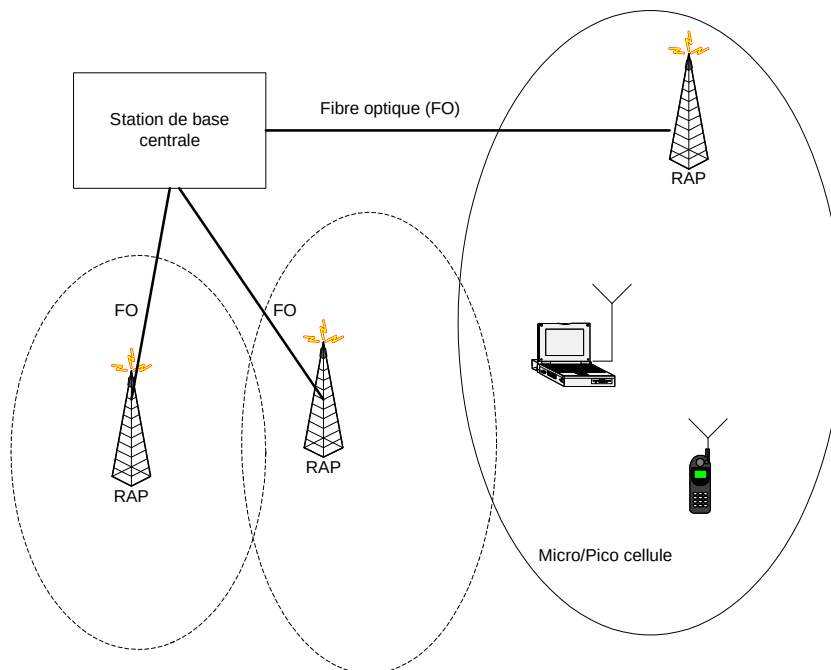


FIG. 5.1 – Architecture d'un réseau sans fil avec utilisation de la fibre optique

Le système de transmission de la liaison ascendante est tel que le signal radio-fréquence émis par la station mobile est reçu par une antenne à laquelle est associée un gain complexe. Le signal reçu est ensuite converti en un signal optique via une diode laser. En régime permanent, la puissance du signal en sortie du laser est une fonction non-linéaire du signal radio fréquence reçu. En notant $s(t)$ le signal radio-fréquence on a :

$$P_{optique} = P_0[1 + ms(t) + A_2m^2s^2(t) + A_3m^3s^3(t)]$$

où m est l'indice de modulation optique et A_2 et A_3 sont des coefficients de non-linéarité qui dépendent des composants utilisés. Le signal optique est ensuite transmis à travers une fibre optique à mode unique. La fonction de transfert d'une fibre optique à mode unique définie par l'UIT-T¹ est :

$$H(f) = e^{-j\alpha L(f-f_0)^2}$$

où α est le coefficient de dispersion, f_0 la fréquence de la porteuse optique et L la longueur de la fibre. Quant à la conversion de l'optique vers l'électrique, elle est régie par les équations classiques d'un système Laser.

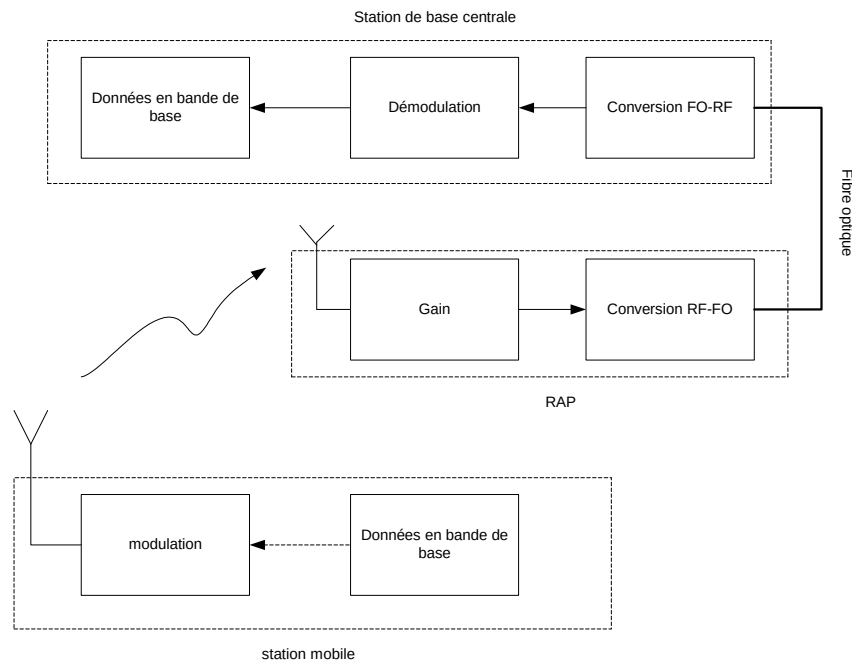


FIG. 5.2 – Schéma fonctionnel d'une liaison ascendante

Quant à la transmission de la liaison descendante, elle est le dual de la liaison ascendante.

Lorsque la longueur de la fibre est relativement courte (quelques kilomètres) et la fréquence radio de quelques GHz, les effets de dispersion de la fibre et le bruit inhérent au laser sont négligeables [136]. Dans ce cas, la distorsion non-linéaire introduite par la conversion du signal électrique en un signal optique devient la plus importante [38, 39]. Le canal caractérisant la liaison ascendante peut donc être représenté par un modèle de Wiener dans lequel le filtre linéaire représente le canal de communication sans fil et la non-linéarité statique la distorsion non-linéaire. De même, la liaison descendante, vue du côté de la station mobile peut être représentée par un modèle de Hammerstein.

¹Commission technique de l'UIT, l'Union Internationale des Télécommunications.

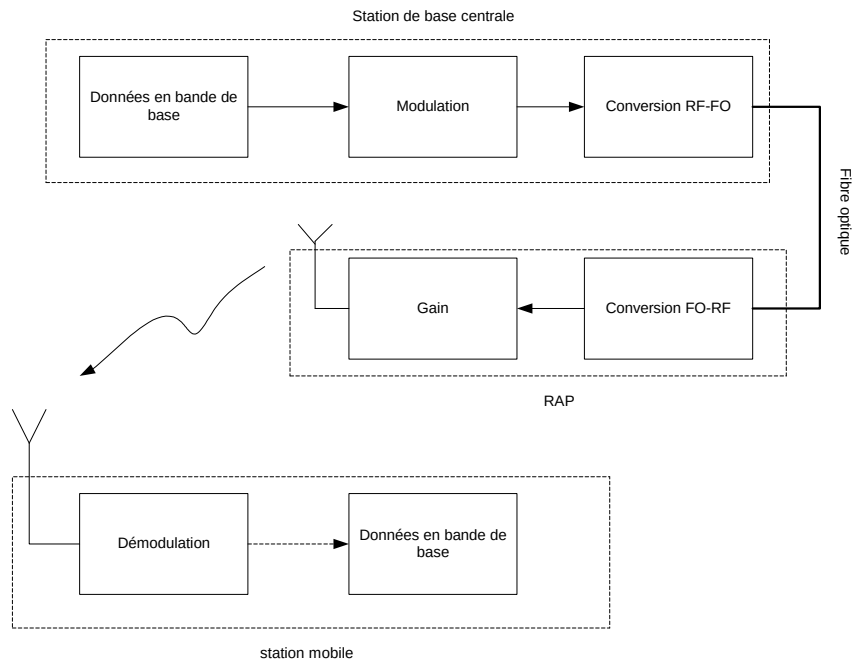


FIG. 5.3 – Schéma fonctionnel d'une liaison descendante

5.2.1 Système multi-capteurs

La liaison descendante est essentiellement mono-capteur puisqu'en général les stations mobiles ne sont dotées que d'une seule antenne. Dans la configuration de la figure 5.2, la liaison ascendante est représentée avec un seul capteur. Mais en fait, plusieurs capteurs peuvent-être utilisés, chacun étant caractérisé par un gain complexe donné. Nous pouvons considérer deux cas :

- Chaque capteur est associé à une liaison optique donnée. Nous avons donc autant de polynômes statiques que de capteurs (Figure 5.4).
- Tous les capteurs utilisent la même liaison optique via un multiplexage puis un démultiplexage. Il n'y a donc qu'un seul polynôme statique (Figure 5.5).

Outre l'accès au réseau sans fil, nous pouvons aussi considérer comme seconde application, un système d'antennes distribuées pour la gestion des communications d'un immeuble. Prenons le cas d'une société qui a son réseau de communication interne. Plusieurs antennes permettent la réception des communications via un réseau sans fil et une fibre optique permet de distribuer les communications dans l'immeuble ou dans un domaine donné. Dans ce cas, la solution la plus économique est de recourir au multiplexage, donc à la deuxième configuration.

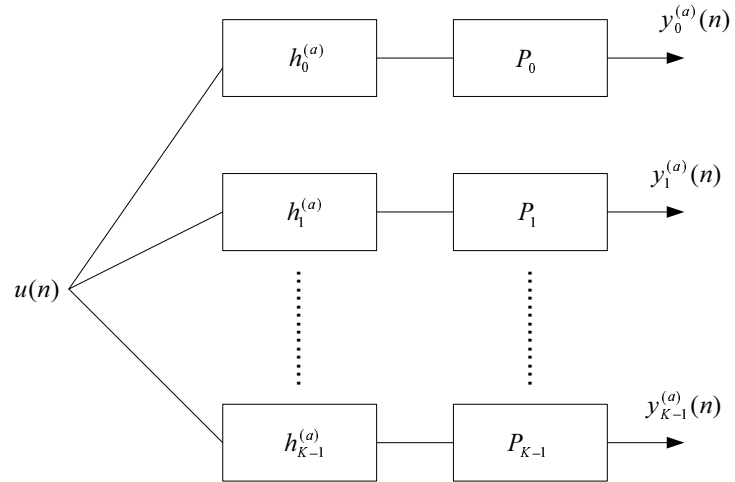


FIG. 5.4 – Configuration ayant plusieurs liaisons optiques

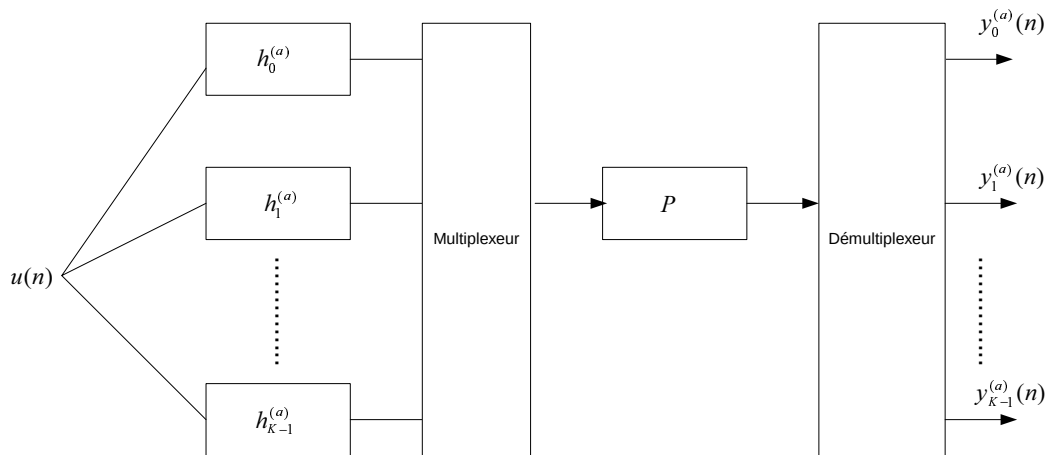


FIG. 5.5 – Configuration ayant une liaison optique partagée

5.2.2 Modélisation

Le canal entre la station mobile et le $(k + 1)$ -ème capteur est modélisé par un filtre RIF de réponse impulsionnelle $h_k^{(a)}$. Sous l'hypothèse que la source est à bande étroite par rapport aux antennes de réception, ce qui est vérifié dans le cas des standards GSM et UMTS, la réponse impulsionnelle du canal de transmission $h_k^{(a)}$ s'écrit alors :

$$h_k^{(a)}(i) = \alpha_k h(i)$$

où α_k est le gain complexe de l'antenne considérée [30].

Le signal reçu par le capteur passe ensuite au travers d'une non-linéarité statique. Le modèle global de la liaison ascendante est donc représenté par un modèle de Wiener. Celui-ci correspond à un modèle de Volterra à noyaux séparables tel que :

$$h_{p,k}^{(a)}(i_1, \dots, i_p) = c_{p,k} \prod_{j=1}^p h_k^{(a)}(i_j)$$

où les $c_{p,k}$ sont les coefficients du polynôme P_k . On a alors :

$$\begin{aligned} h_{p,k}^{(a)}(i_1, \dots, i_p) &= c_{p,k} \prod_{j=1}^p (\alpha_k h(i_j)) \\ &= c_{p,k} \alpha_k^p \prod_{j=1}^p h(i_j) \\ &= \beta_{k,p} \prod_{j=1}^p h(i_j) \end{aligned} \quad (5.1)$$

avec

$$\beta_{k,p} = c_{p,k} \alpha_k^p$$

Dans le contexte de la seconde configuration, i.e. lorsqu'un multiplexage est utilisé, nous avons :

$$\beta_{k,p} = c_p \alpha_k^p$$

La relation entrée-sortie de la liaison ascendante peut-donc s'écrire comme :

$$y_k^{(a)}(n) = \sum_{p=1}^P \sum_{i_1=0}^M \dots \sum_{i_p=i_{p-1}}^M h_{p,k}^{(a)}(i_1, \dots, i_p) \prod_{j=1}^p u(n - i_j) \quad (5.2)$$

$h_{p,k}^{(a)}(i_1, \dots, i_p)$ étant donné par (5.1).

Dans le cas mono-capteur nous avons :

$$y^{(a)}(n) = \sum_{p=1}^P \sum_{i_1=0}^M \dots \sum_{i_p=i_{p-1}}^M h_p^{(a)}(i_1, \dots, i_p) \prod_{j=1}^p u(n - i_j) \quad (5.3)$$

où

$$h_p^{(a)}(i_1, \dots, i_p) = \beta_p \prod_{j=1}^p h(i_j), \quad \beta_p = c_p \alpha^p \quad (5.4)$$

Quant à la liaison descendante, elle est représentée par le modèle de Hammerstein suivant :

$$y^{(d)}(n) = \sum_{p=1}^P \sum_{i=0}^M h_p^{(d)}(i) u^p(n-i) \quad (5.5)$$

5.3 Identification et égalisation semi-aveugle de la liaison descendante

La liaison descendante est représentée par le modèle SISO (5.5). La méthode que nous proposons consiste à construire une représentation de type SIMO à partir du SISO en un modèle SIMO via l'utilisation d'une matrice de redondance. Certaines branches du modèle SIMO résultant, se réduisent à des sous-canaux représentés par un modèle polynomial instantané. Toute l'information sur le canal SISO est contenue dans ces sous-canaux polynômiaux. Leur identification induit donc celle du modèle SISO original. Nous considérons une transmission en bloc de données dont le premier élément est le signal utile au processus d'identification. Celui-ci est un symbole pris dans une séquence pilote donnée. Dans un premier temps, nous allons montrer comment exploiter la matrice de redondance pour passer d'un modèle SISO à un modèle SIMO. Puis, nous considérerons la question de l'estimation d'un modèle polynomial instantané.

5.3.1 Transformation du modèle de Hammerstein en un modèle SIMO polynomial instantané

Depuis quelques années, la communauté du traitement du signal s'intéressant à la problématique de l'identification aveugle des canaux linéaires, s'est enrichie de l'approche basée sur l'exploitation des statistiques du second ordre des sorties du canal obtenues soit par sur-échantillonnage soit en utilisant plusieurs antennes. Cette approche a permis le développement de nouvelles techniques d'identification aveugle moins coûteuses en complexité de calcul que les méthodes basées sur l'exploitation des statistiques d'ordre supérieur. Cependant certaines limitations ont été mises en évidence [32]. Il a donc été proposé de recourir à des émetteurs qui induisent une cyclostationnarité du signal émis [46]. Dans la même optique, il a été montré qu'en introduisant une certaine redondance dans le bloc de données émises, le même résultat peut être obtenu [26]. L'écueil de ces approches est la réduction du taux d'information transmise. Suivant certaines conditions [26], cette réduction peut cependant être limitée.

Dans le cas non linéaire, les statistiques du second-ordre ont surtout été exploitées en vue de l'égalisation pour des canaux SIMO [113, 78]. Dans [79] un précodeur est utilisé

pour égaliser un canal SIMO non linéaire. Toujours dans le cadre de l'égalisation, [116] ont aussi utilisé un précodeur pour l'implémentation d'un égaliseur supervisé.

Dans notre étude, nous considérons un canal SISO. En utilisant une transmission en bloc et une matrice de redondance, nous montrons comment ce modèle SISO peut être vu comme un modèle SIMO. Cette structure sera ensuite exploitée pour identifier le canal.

Considérons le bloc d'entrées défini par le vecteur $U(n)$ de dimensions $L \times 1$, L étant la taille du bloc :

$$U(n) = \begin{pmatrix} u(nL) & u(nL-1) & \cdots & u(nL-L+1) \end{pmatrix}^T \quad (5.6)$$

En utilisant une matrice de redondance $\mathbf{C}$ de dimension $K \times L$ ($K > L$) nous pouvons générer le bloc de données transformées $X(n)$ de dimensions $K \times 1$ suivant :

$$X(n) = \mathbf{C}U(n) = \begin{pmatrix} x(nK) & x(nK-1) & \cdots & x(nK-K+1) \end{pmatrix}^T$$

$X(n)$ est un bloc de K données formé à partir du bloc (5.6) de L données informatives. Le taux d'information contenu dans $X(n)$ a donc été réduit de L/K . On peut alors définir un ratio de redondance comme $(K-L)/K$. Une augmentation de ce ratio réduit le taux d'information. Dans [26], il a été prouvé qu'une matrice de redondance assurant un ratio de redondance minimal ² est :

$$\mathbf{C} = \begin{pmatrix} C_M & 0_{(2M+1) \times (L-1)} \\ 0_{(L-1) \times 1} & I_{(L-1) \times (L-1)} \end{pmatrix} \quad (5.7)$$

où C_M est un vecteur de dimension $(2M+1) \times 1$ défini par :

$$C_M = \begin{pmatrix} \underbrace{0 \cdots 0}_M 1 \underbrace{0 \cdots 0}_M \end{pmatrix}^T \quad (5.8)$$

Les dimensions des blocs avant et après transformation par la matrice $\mathbf{C}$, i.e. L et K respectivement, sont reliées par :

$$K = L + 2M \quad (5.9)$$

La matrice $\mathbf{C}$ n'étant constituée que par des 0 et des 1, la transformation des données ne consiste donc qu'en une insertion de $2M$ zéros entre les données. Plus exactement, M zéros avant et après le premier symbole du bloc $U(n)$:

$$X(n) = \begin{pmatrix} \underbrace{0 \cdots 0}_M u(nL) \underbrace{0 \cdots 0}_M u(nL-1) \cdots u(nL-L+1) \end{pmatrix}^T \quad (5.10)$$

Nous pouvons aussi noter les éléments de $X(n)$ comme suit :

$$x(nK-i) = \begin{cases} 0 & \text{si } i = 0, \dots, M-1 \text{ ou } i = M+1, \dots, 2M \\ u(nL) & \text{si } i = M \\ u(nL-i+2M) & \text{si } i = 2M+1, \dots, K-1 \end{cases} \quad (5.11)$$

²dans le sens où les conditions d'identifiabilité aveugle des canaux linéaires sont satisfaites

Soit le bloc de données en sortie $Y^{(d)}(n)$ de dimensions $K \times 1$ suivant

$$Y^{(d)}(n) = (y^{(d)}(nK) \ y^{(d)}(nK-1) \ \dots \ y^{(d)}(nK-K+1))^T$$

Ses éléments sont donnés par :

$$y^{(d)}(nK-i) = \sum_{p=1}^P \sum_{j=0}^M h_p^{(d)}(j)x^p(nK-i-j) + w(nK-i) \quad (5.12)$$

où w est le bruit additif.

Selon (5.11), $x(nK-i-j)$ n'est non nul que si $i+j = M$ ou $i+j \geq 2M+1$. Par conséquent l'équation (5.12) s'écrit :

$$y^{(d)}(nK-i) = \sum_{p=1}^P h_p^{(d)}(M-i)x^p(nK-M) + \sum_{p=1}^P \sum_{j=2M-i+1}^M h_p^{(d)}(j)x^p(nK-i-j) + w(nK-i)$$

ou encore :

$$y^{(d)}(nK-i) = \sum_{p=1}^P h_p^{(d)}(M-i)x^p(nK-M) + \sum_{p=1}^P \sum_{t=2M+1}^{M+i} h_p^{(d)}(t-i)x^p(nK-t) + w(nK-i)$$

Sachant aussi que $h_p(j) = 0$ si $j < 0$ ou si $j > M$, nous avons alors

$$y^{(d)}(nK-i) = \begin{cases} \sum_{p=1}^P h_p^{(d)}(M-i)x^p(nK-M) + w(nK-i) & \text{si } i = 0, \dots, M \\ \sum_{p=1}^P \sum_{t=2M+1}^{M+i} h_p^{(d)}(t-i)x^p(nK-t) + w(nK-i) & \text{si } i = M+1, \dots, K-1 \end{cases}$$

Dans l'équation ci-dessus, lorsque $t \geq K$ les symboles $x(nK-t)$ n'appartiennent pas au bloc $X(n)$ mais plutôt aux blocs précédents. Pour éviter toute interférence entre blocs, nous devons prendre $t \leq M+i < K$ ce qui équivaut à prendre $i = 0, \dots, K-M-1$. Nous considérons donc le bloc tronqué $\bar{Y}^{(d)}(n)$, de dimensions $(K-M) \times 1$, suivant :

$$\bar{Y}^{(d)}(n) = (y^{(d)}(nK) \ y^{(d)}(nK-1) \ \dots \ y^{(d)}(nK-K+M+1))^T$$

dont les éléments sont donnés par :

$$y^{(d)}(nK-i) = \begin{cases} \sum_{p=1}^P h_p^{(d)}(M-i)u^p(nL) + w(nK-i) & \text{si } i = 0, \dots, M \\ \sum_{p=1}^P \sum_{t=2M+1}^{M+i} h_p^{(d)}(t-i)u^p(nL-t+2M) + w(nK-i) & \text{si } i = M+1, \\ & \dots, K-M-1 \end{cases}$$

Nous pouvons donc remarquer que les $(M+1)$ premiers éléments du bloc $\bar{Y}^{(d)}(n)$ sont donnés par la relation suivante

$$y^{(d)}(nK-i) = \sum_{p=1}^P h_p^{(d)}(M-i)u^p(nL) + w(nK-i), \ i = 0, \dots, M \quad (5.13)$$

Quant aux $(L - 1)$ derniers éléments, ils sont le fruit du mélange convolutif et non-linéaire des $(L - 1)$ dernières composantes du bloc $U(n)$:

$$y^{(d)}(nK - i) = \sum_{p=1}^P \sum_{j=1}^{i-M} h_p^{(d)}(2M - i + j) u^p(nL - j) + w(nK - i), i = M + 1, \dots, K - M - 1 \quad (5.14)$$

Posons :

$$\mathcal{H}_p^{(d)} = \begin{pmatrix} h_p^{(d)}(0) & \dots & h_p^{(d)}(M) & \dots & 0 \\ & \ddots & & \ddots & \\ & & h_p^{(d)}(0) & \dots & h_p^{(d)}(M) \end{pmatrix} \quad (5.15)$$

$\mathcal{H}_p^{(d)}$ est une matrice de dimension $(K - M) \times K$. Nous définissons aussi la matrice du canal :

$$\mathcal{H}^{(d)} = \begin{pmatrix} \mathcal{H}_1^{(d)} & \dots & \mathcal{H}_P^{(d)} \end{pmatrix} \quad (5.16)$$

de dimensions $(K - M) \times PK$. Par suite, la représentation vectorielle de la relation entrée-sortie du système est :

$$\bar{Y}^{(d)}(n) = \mathcal{H}^{(d)} \bar{X}(n) + \bar{W}(n) \quad (5.17)$$

où $\bar{X}(n)$ et $\bar{W}(n)$ sont des vecteurs de dimensions respectives $PK \times 1$ et $(K - M) \times 1$ définis par

$$\bar{X}(n) = \begin{pmatrix} X^T(n) & X^T(n) \circ X^T(n) & \dots & \underbrace{X^T(n) \circ X^T(n) \dots \circ X^T(n)}_{P \text{ termes}} \end{pmatrix}^T$$

○ définissant le produit de Hadamard et

$$\bar{W}(n) = \begin{pmatrix} w(nK) & w(nK - 1) & \dots & w(nK - K + M + 1) \end{pmatrix}^T$$

En ne s'intéressant qu'aux $(M + 1)$ premières composantes de $\bar{Y}^{(d)}$, i.e. aux éléments donnés par $y^{(d)}(nK - i)$, $i = 0, \dots, M$, la relation (5.13) peut donc être vue comme étant la relation d'entrée/sortie d'un filtre polynomial statique ayant comme entrée $u(nL)$. Sachant que $i = 0, \dots, M$, nous avons donc un filtre SIMO d'entrée $u(nL)$, ayant $M + 1$ sorties. On peut remarquer que tous les coefficients du modèle SISO se retrouvent dans ce modèle SIMO. En identifiant le modèle SIMO, on identifie donc le modèle SISO de départ.

Par ailleurs, on peut aussi remarquer que les échantillons de l'entrée ayant permis de générer les $(M + 1)$ premières composantes de $\bar{Y}^{(d)}(n)$ sont différents de ceux ayant permis de générer ses $(L - 1)$ dernières composantes. Nous pouvons donc choisir $u(nL)$ comme étant le symbole d'une séquence pilote non informative alors que les $u(nL - j)$, $j = 1, \dots, L - 1$, contiennent l'information utile. Nous pouvons donc nous concentrer sur l'identification du modèle SIMO résultant en ne tenant compte que de la séquence pilote. Une fois le modèle du canal identifié, il servira à l'estimation des symboles informatifs.

Exemple 5.1 En guise d'illustration, considérons, dans le cas sans bruit, un modèle d'Hammerstein d'ordre trois, de mémoire $M = 2$. Prenons des blocs tels que $K = 7$ et $L = 3$. La matrice du canal est : $\mathcal{H}^{(d)} = \left(\mathcal{H}_1^{(d)} \mathcal{H}_2^{(d)} \mathcal{H}_3^{(d)} \right)$ avec :

$$\mathcal{H}_p^{(d)} = \begin{pmatrix} h_p^{(d)}(0) & h_p^{(d)}(1) & h_p^{(d)}(2) & 0 & 0 & 0 & 0 \\ 0 & h_p^{(d)}(0) & h_p^{(d)}(1) & h_p^{(d)}(2) & 0 & 0 & 0 \\ 0 & 0 & h_p^{(d)}(0) & h_p^{(d)}(1) & h_p^{(d)}(2) & 0 & 0 \\ 0 & 0 & 0 & h_p^{(d)}(0) & h_p^{(d)}(1) & h_p^{(d)}(2) & 0 \\ 0 & 0 & 0 & 0 & h_p^{(d)}(0) & h_p^{(d)}(1) & h_p^{(d)}(2) \end{pmatrix}, \quad p = 1, 2, 3$$

On a par ailleurs :

$$\begin{aligned} U(n) &= \begin{pmatrix} u(3n) & u(3n-1) & u(3n-2) \end{pmatrix}^T \\ X(n) &= \begin{pmatrix} 0 & 0 & u(3n) & 0 & 0 & u(3n-1) & u(3n-2) \end{pmatrix}^T \\ \bar{X}(n) &= \begin{pmatrix} X^T(n) & X^T \circ X^T & X^T \circ X^T \circ X^T \end{pmatrix}^T \end{aligned}$$

La relation (5.17) s'écrit alors dans le cas sans bruit :

$$\bar{Y}^{(d)}(n) = \begin{pmatrix} \frac{y^{(d)}(7n)}{y^{(d)}(7n-2)} \\ \frac{y^{(d)}(7n-1)}{y^{(d)}(7n-4)} \end{pmatrix} = \begin{pmatrix} \frac{h_1(2)u(3n) + h_2(2)u^2(3n) + h_3(2)u^3(3n)}{h_1(1)u(3n) + h_2(1)u^2(3n) + h_3(1)u^3(3n)} \\ \frac{h_1(0)u(3n) + h_2(0)u^2(3n) + h_3(0)u^3(3n)}{h_1(2)u(3n-1) + h_2(2)u^2(3n-1) + h_3(2)u^3(3n-1)} \end{pmatrix}$$

Le principe de transformation étant posé, nous allons montrer dans la section qui suit comment identifier chaque sous-canal.

5.3.2 Identification d'un modèle SIMO polynomial statique

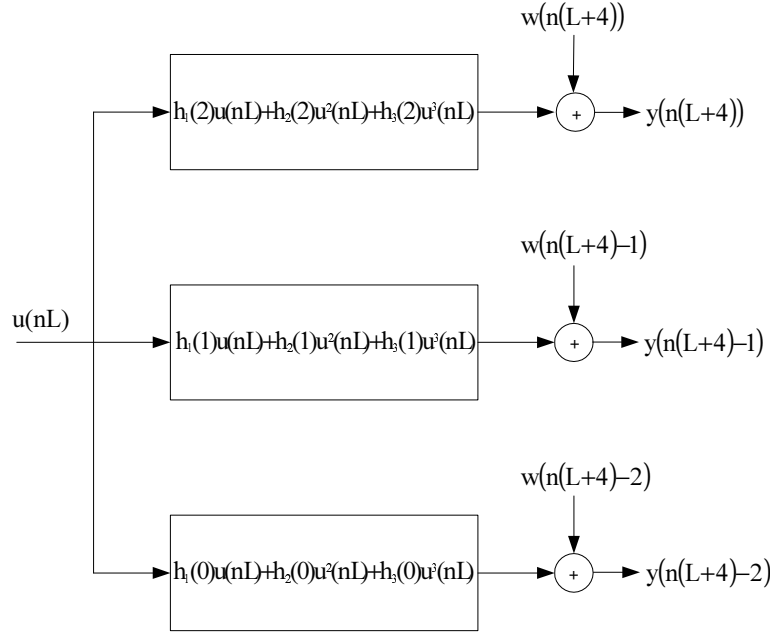
Dans la littérature, deux algorithmes d'identification aveugle des modèles polynômiaux sans mémoire ont été proposés dans le cas SISO. Le premier ne s'applique que dans le cas des signaux réels sinusoïdaux [81] tandis que le second minimise par une méthode du gradient un critère d'ajustement de cumulants d'ordre quatre [103]. Signalons que dans ce dernier cas la convergence de l'algorithme n'est garantie que vers un minimum local.

Rappelons le modèle à estimer (5.13) :

$$y^{(d)}(nK - i) = \sum_{p=1}^P h_p^{(d)}(M - i) u^p(nL) + w(nK - i), \quad i = 0, \dots, M$$

L'objectif de cette section est d'estimer les vecteurs $\Theta_i^{(d)}$ définis par :

$$\Theta_i^{(d)} = \begin{pmatrix} h_1^{(d)}(M - i) & \dots & h_P^{(d)}(M - i) \end{pmatrix}^T$$


 FIG. 5.6 – Cas d'un modèle de Hammerstein d'ordre 3, ($M = 2$)

5.3.2.1 Estimation au sens des moindres carrés d'un modèle polynomial statique

Dans une approche d'identification semi-aveugle, la séquence pilote $u(nL)$, $n = 0, \dots, N-1$ est connue du récepteur, notre objectif est de trouver un vecteur $\Theta_i^{(d)}$ tel que :

$$\hat{\Theta}_i^{(d)} = \arg \min_{\Theta_i^{(d)}} \|Y_i^{(d)} - \mathbf{V}\Theta_i^{(d)}\|^2 \quad (5.18)$$

où $Y_i^{(d)}$ est un vecteur de dimensions $N \times 1$, N étant le nombre de blocs de mesures, défini par :

$$Y_i^{(d)} = (y^{(d)}(-i) \quad \dots \quad y^{(d)}((N-1)K-i))^T$$

et $\mathbf{V}$ est une matrice de Vandermonde de dimensions $N \times P$ telle que :

$$\mathbf{V} = \begin{pmatrix} u(0) & \dots & u^{P-1}(0) & u^P(0) \\ u(L) & \dots & u^{P-1}(L) & u^P(L) \\ \vdots & \dots & \vdots & \vdots \\ u((N-1)L) & \dots & u^{P-1}((N-1)L) & u^P((N-1)L) \end{pmatrix} \quad (5.19)$$

Cette méthode requiert la connaissance des signaux $u(nL)$, $n = 0, \dots, N-1$ et l'infériorité du degré de non-linéarité P par rapport au nombre N d'échantillons. Condition qui n'est pas trop restrictive d'autant plus que les polynômes considérés sont généralement d'ordre faible, i.e. ($P \leq 5$). Dans le cas où $u(nL)$ prend M_u valeurs distinctes, il est nécessaire d'avoir $M_u > P+1$.

Le vecteur de paramètres optimal est donné par :

$$\hat{\Theta}_i^{(d)} = (\mathbf{V}^T \mathbf{V})^{-1} \mathbf{V}^T Y_i^{(d)}$$

Afin de simplifier la procédure d'estimation, en évitant l'inversion d'une matrice pleine, nous considérons la décomposition QR de la matrice de Vandermonde $\mathbf{V}$:

$$\mathbf{V} = \mathbf{Q}\mathbf{R} \quad (5.20)$$

où $\mathbf{Q}$ est une matrice orthogonale de même dimension que $\mathbf{V}$ et $\mathbf{R}$ une matrice triangulaire supérieure. Tenant compte de cette factorisation, nous obtenons :

$$\mathbf{R}\hat{\Theta}_i^{(d)} = \mathbf{Q}^T \mathbf{Y}_i^{(d)} \quad (5.21)$$

$\mathbf{Q}^T \mathbf{Y}_i^{(d)}$ étant un vecteur de dimension P , les coefficients de $\hat{\Theta}_i^{(d)}$ sont alors obtenus en résolvant un système d'équations triangulaire.

Remarque 5.1 : Il est à noter que dans le cas de notre modèle SIMO, le système d'équations (5.21) est à résoudre pour $i = 0, \dots, M$. Cependant on notera que seul le vecteur $\mathbf{Y}_i^{(d)}$ change. La matrice de Vandermonde $\mathbf{V}$ est commune à tous les sous-canaux. En conséquence, la factorisation QR (5.20) ne s'effectue qu'une seule fois.

La procédure peut donc se résumer comme suit :

Algorithme 5.1 Identification au sens des moindres carrés

1. A partir des mesures $u(nL)$ et $y^{(d)}(nK - i)$, $n = 0, \dots, N - 1$ et $i = 0, \dots, M$, construire la matrice $\mathbf{V}$ et les vecteurs $\mathbf{Y}_i^{(d)}$;
2. Effectuer la factorisation QR de la matrice $\mathbf{V}$;
3. Pour chaque i résoudre le système d'équations (5.21).

Remarque 5.2 : L'erreur moyenne entre les sorties réelles $\mathbf{Y}_i^{(d)}$ et celles générées par le modèle estimé est calculée à partir du vecteur $\mathbf{E} = \mathbf{Y}_i^{(d)} - \mathbf{V}\hat{\Theta}_i^{(d)}$. Si l'ordre P du modèle n'est pas connu a priori, il peut être choisi en minimisant $\|\mathbf{E}\|^2$. Il suffit pour cela d'appliquer l'algorithme 5.1 de manière itérative par rapport à l'ordre P jusqu'à ce que $\|\mathbf{E}\|^2$ soit relativement constante.

Remarque 5.3 : Puisque l'identification de chaque branche, i.e. l'estimation de chaque vecteur $\Theta_i^{(d)}$, se fait connaissant la séquence pilote, un recours à une procédure adaptative est possible. Une telle procédure peut servir dans le cas où les caractéristiques du canal varient dans le temps. Si l'estimé du canal sert à implémenter l'égaliseur du signal informatif, on obtient un égaliseur adaptatif qui suit les variations du canal.

5.3.2.2 Utilisation d'une séquence pilote périodique

Dans le développement de la section précédente, nous avons supposé que la séquence $u(nL)$ est connue. Supposons que $u(nL)$ est une séquence pilote périodique de période $T \geq P$, indépendante du signal informatif. Un choix possible est :

$$u(nL) = e^{j2\pi n/T}, \quad n = 0, 1, \dots, T - 1$$

Une telle séquence a été proposée par [33] comme séquence pilote pour l'identification semi-aveugle de canaux de communications linéaires.

En considérant le cas sans bruit, la relation (5.13) peut donc s'écrire :

$$y^{(d)}(nK - i) = \sum_{p=1}^P h_p^{(d)}(M - i) e^{j\frac{2\pi}{T}pn}$$

Posons $e_p(n) = e^{j\frac{2\pi}{T}pn}$. En définissant le produit scalaire suivant [99] :

$$\langle e_p, e_q \rangle = \frac{1}{T} \sum_{n=0}^{T-1} e_p(n) e_q^*(n)$$

on peut remarquer l'orthogonalité des séquences $e_p(n)$, $p = 1, \dots, P$, $n = 0, \dots, T - 1$, i.e.

$$\langle e_p, e_q \rangle = \begin{cases} 1, & \text{si } p = q \\ 0, & \text{sinon} \end{cases}$$

Pour $q = 1, \dots, P$, on a donc :

$$\begin{aligned} \langle y^{(d)}(nK - i), e_q \rangle &= \frac{1}{T} \sum_{n=0}^{T-1} y^{(d)}(nK - i) e_q^*(n) \\ &= \frac{1}{T} \sum_{n=0}^{T-1} \sum_{p=1}^P h_p^{(d)}(M - i) e_p(n) e_q^*(n) \\ &= \sum_{p=1}^P h_p^{(d)}(M - i) \left(\frac{1}{T} \sum_{n=0}^{T-1} e_p(n) e_q^*(n) \right) \\ &= h_q^{(d)}(M - i) \end{aligned} \quad (5.22)$$

Autrement dit les coefficients du modèle polynômial statique sont donnés par :

$$h_q^{(d)}(M - i) = \frac{1}{T} \sum_{n=0}^{T-1} y^{(d)}(nK - i) e^{-j\frac{2\pi}{T}qn}, \quad q = 1, \dots, P, \quad i = 0, \dots, M \quad (5.23)$$

Cette méthode est très élégante et simple en terme de complexité d'implémentation. Théoriquement, la connaissance *a priori* de l'ordre P n'est pas nécessaire. L'estimation des coefficients $h_q^{(d)}(M - i)$ étant découplée, si P est l'ordre effectif, alors $h_q^{(d)}(M - i) = 0$, si $q > P$. Cependant, il est nécessaire de signaler que ce développement a été effectué dans le cas sans bruit. Nous pouvons dès lors remarquer que l'estimation des $h_q^{(d)}(M - i)$ est biaisée par la présence du bruit. Ce biais est égal à :

$$b_q = \frac{1}{T} \sum_{n=0}^{T-1} w(nK - i) e^{-j\frac{2\pi}{T}qn}$$

Algorithme 5.2 Identification à l'aide d'une séquence pilote périodique

1. Enregistrer les mesures $y^{(d)}(nK - i)$, $n = 0, \dots, T - 1$ et $i = 0, \dots, M$
2. Déterminer les paramètres du modèle $h_q^{(d)}(M - i)$, $q = 1, \dots, P$ en utilisant la relation (5.23).

5.3.3 Simulation

Considérons l'identification du système suivant :

$$\begin{aligned} y^{(d)}(n) = & u(n) + 0.45u(n-1) + 0.35u(n-2) \\ & + 0.3u^2(n) - 0.6u^2(n-2) \\ & + 0.1u^3(n) - 0.01u^3(n-1) + w(n) \end{aligned} \quad (5.24)$$

L'entrée du système est un signal *PAM* – 4. Nous considérons la transmission de blocs de taille $L = 4$. On a $M = 2$ et donc, selon (5.9), $K = 8$. Par suite, les branches SIMO nécessaires à la procédure d'identification sont telles que :

$$\begin{aligned} y^{(d)}(nK) &= u(nL) + 0.3u^2(nL) + 0.1u^3(nL) + w(nK) \\ y^{(d)}(nK-1) &= 0.45u(nL) - 0.01u^3(nL) + w(nK-1) \\ y^{(d)}(nK-2) &= 0.35u(nL) - 0.6u^2(nL) + w(nK-2) \end{aligned}$$

En vue de comparer la précision des différentes méthodes, nous définissons aussi l'erreur quadratique moyenne d'estimation normalisée (EQMEN) :

$$EQMEN = \frac{\sum_{p=1}^P \sum_{i=0}^M (h_p^{(d)}(i) - \hat{h}_p^{(d)}(i))^2}{\sum_{p=1}^P \sum_{i=0}^M (h_p^{(d)}(i))^2}$$

où $h_p^{(d)}(i)$ sont les coefficients du système à identifier et $\hat{h}_p^{(d)}(i)$ les coefficients estimés.

5.3.3.1 Estimation au sens des moindres carrés

Nous utilisons deux types de séquence pilote. La première séquence est constituée par les symboles de tête pris de manière aléatoire dans un alphabet de même type que celui du signal informatif tandis que la seconde suit une distribution gaussienne. Pour différents niveaux de bruit et pour 100 expériences indépendantes nous avons les résultats des tableaux 5.1 et 5.2. Il est à noter que l'identification a été réalisée en considérant la transmission successive de 50 blocs de données, chaque bloc n'ayant qu'un seul symbole de la séquence pilote.

5.3. Identification et égalisation semi-aveugle de la liaison descendante

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_1(0) = 1$	1.001 ± 0.013	0.994 ± 0.208	1.022 ± 0.365
$h_1(1) = 0.45$	0.450 ± 0.005	0.462 ± 0.079	0.460 ± 0.140
$h_1(2) = 0.35$	0.350 ± 0.006	0.348 ± 0.092	0.331 ± 0.157
$h_2(0) = 0.3$	0.300 ± 0.008	0.295 ± 0.084	0.281 ± 0.154
$h_2(2) = -0.6$	-0.599 ± 0.004	-0.601 ± 0.047	-0.608 ± 0.077
$h_3(0) = 0.1$	0.100 ± 0.015	0.108 ± 0.217	0.082 ± 0.369
$h_3(1) = -0.01$	-0.010 ± 0.006	-0.019 ± 0.088	-0.015 ± 0.146

TAB. 5.1 – Estimation paramétrique en utilisant une séquence PAM-4

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_1(0) = 1$	0.999 ± 0.014	1.008 ± 0.122	1.013 ± 0.168
$h_1(1) = 0.45$	0.450 ± 0.003	0.443 ± 0.034	0.462 ± 0.056
$h_1(2) = 0.35$	0.349 ± 0.009	0.353 ± 0.050	0.334 ± 0.121
$h_2(0) = 0.3$	0.299 ± 0.004	0.311 ± 0.059	0.296 ± 0.104
$h_2(2) = -0.6$	-0.600 ± 0.003	-0.604 ± 0.029	-0.596 ± 0.077
$h_3(0) = 0.1$	0.100 ± 0.003	0.097 ± 0.047	0.096 ± 0.070
$h_3(1) = -0.01$	-0.010 ± 0.001	-0.008 ± 0.012	-0.013 ± 0.023

TAB. 5.2 – Estimation paramétrique en utilisant une séquence gaussienne

Nous pouvons noter qu'en moyenne les valeurs estimées des paramètres du canal correspondent bien aux valeurs réelles. L'effet du bruit est visible sur l'écart-type autour des valeurs moyennes. Plus le niveau de bruit est important plus l'écart-type est grand. Par ailleurs, on remarque aussi que les écarts-types sont plus restreints lorsque la séquence pilote est une gaussienne plutôt qu'un signal PAM-4. Ceci vient du fait que dans le cas d'une séquence gaussienne, le signal est beaucoup plus riche. Le signal PAM-4 n'ayant que 4 symboles, dans le cas sans bruit, il n'y a que 4 équations différentes possibles. Augmenter le nombre de mesures n'apporte donc aucune amélioration dans l'estimation du modèle. Dans le cas gaussien, par contre, le fait d'augmenter le nombre de mesures apporte des équations supplémentaires au système d'équations permettant ainsi d'affiner l'estimation du modèle.

5.3.3.2 Utilisation d'une séquence pilote périodique

A présent nous allons recourir à une séquence périodique $u(n) = e^{j2\pi n/T}$ de période $T = 5$. En considérant la transmission successive d'un nombre de blocs de données correspondant à la période de la séquence pilote, i.e. 5 blocs de données, nous obtenons les résultats du tableau 5.3.

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_1(0) = 1$	1.002 ± 0.019	0.999 ± 0.206	1.041 ± 0.362
$h_1(1) = 0.45$	0.449 ± 0.008	0.453 ± 0.080	0.445 ± 0.150
$h_1(2) = 0.35$	0.349 ± 0.013	0.343 ± 0.130	0.372 ± 0.231
$h_2(0) = 0.3$	0.298 ± 0.020	0.292 ± 0.199	0.207 ± 0.363
$h_2(2) = -0.6$	-0.600 ± 0.013	-0.589 ± 0.143	-0.593 ± 0.217
$h_3(0) = 0.1$	0.102 ± 0.019	0.099 ± 0.206	0.141 ± 0.362
$h_3(1) = -0.01$	-0.010 ± 0.008	-0.007 ± 0.080	-0.015 ± 0.150

TAB. 5.3 – Estimation paramétrique en utilisant une séquence pilote périodique

Signal pilote	SNR		
	5 dB	10 dB	30 dB
Gaussien	5.730×10^{-2}	2.000×10^{-2}	1.756×10^{-4}
PAM-4	2.448×10^{-1}	8.370×10^{-2}	9.774×10^{-4}
Périodique ($T = 5$)	1.230×10^{-1}	4.032×10^{-2}	3.846×10^{-4}

TAB. 5.4 – Evaluation de la mesure EQMEN pour différentes séquences pilotes

L'avantage de l'utilisation d'une séquence pilote périodique est qu'une période de transmission suffit pour l'estimation des paramètres. La méthode d'estimation dans ce cas étant très simple. Par ailleurs, nous pouvons constater que les performances sont en moyenne similaires à celles données par la méthode des moindres carrés pour les autres

types de séquence pilote. La variance des estimées obtenues est légèrement plus importante que dans le cas des moindres carrés avec une séquence PAM-4 et bien plus en comparaison avec la méthode des moindres carrés utilisant une séquence gaussienne. Cette observation est aussi étayée par le tableau 5.4 qui présente l'évaluation de la mesure $EQMEN$. L'estimation est meilleure pour une séquence pilote gaussienne.

Sur la figure 5.7, nous pouvons noter que l'estimation est d'autant plus précise que la période de la séquence pilote augmente.

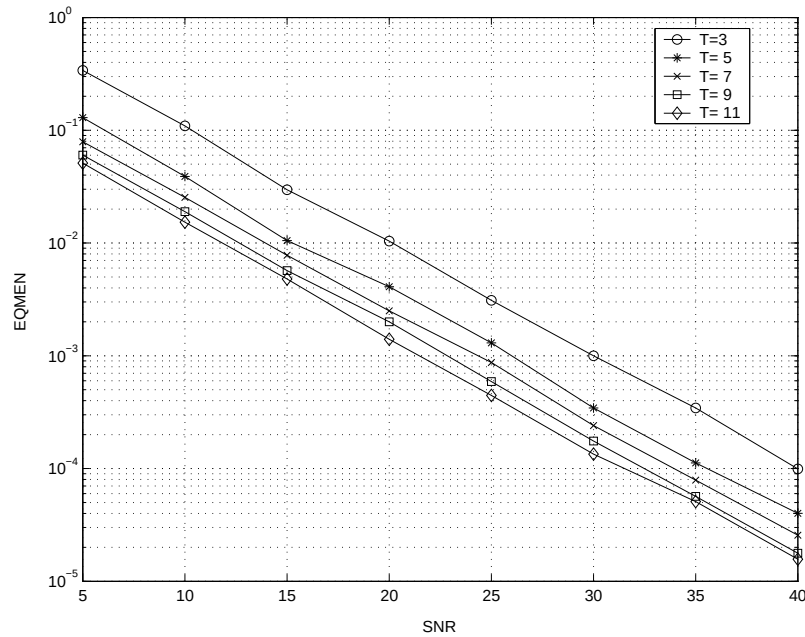


FIG. 5.7 – Evaluation de la précision de l'estimation pour différentes périodes de la séquence pilote

5.3.4 Estimation des symboles transmis

La matrice du canal ayant été déterminée, nous allons nous intéresser à l'estimation des symboles transmis. Pour ce faire, nous allons considérer la partie inférieure de $\bar{Y}^{(d)}(n)$, i.e. celle constituée par les éléments définis par l'équation (5.14), qui contient l'information utile. Deux méthodes conduisant à une égalisation de type forçage à zéro vont être considérées. La première est basée sur la résolution des systèmes d'équations polynômiales tandis que la seconde utilise les inverses des polynômes. Le développement de ces méthodes est fait en considérant le cas sans bruit.

5.3.4.1 Estimation par résolution d'équations polynômiales

Reprenons l'exemple 5.1. La partie inférieure de $\bar{Y}^{(d)}(n)$ est donnée par :

$$\begin{pmatrix} y(7n-3) \\ y(7n-4) \end{pmatrix} = \begin{pmatrix} h_1(2)u(3n-1) + h_2(2)u^2(3n-1) + h_3(2)u^3(3n-1) \\ + w(7n-3) \\ h_1(1)u(3n-1) + h_2(1)u^2(3n-1) + h_3(1)u^3(3n-1) \\ + h_1(2)u(3n-2) + h_2(2)u^2(3n-2) + h_3(2)u^3(3n-2) \\ + w(7n-4) \end{pmatrix}$$

Dans le cas sans bruit nous avons :

$$\begin{pmatrix} y(7n-3) \\ y(7n-4) \end{pmatrix} = \begin{pmatrix} h_1(2)u(3n-1) + h_2(2)u^2(3n-1) + h_3(2)u^3(3n-1) \\ h_1(1)u(3n-1) + h_2(1)u^2(3n-1) + h_3(1)u^3(3n-1) \\ + h_1(2)u(3n-2) + h_2(2)u^2(3n-2) + h_3(2)u^3(3n-2) \end{pmatrix}$$

Nous avons là un système d'équations polynômiales triangulaire à deux inconnues $u(3n-1)$ et $u(3n-2)$ qui sont les symboles informatifs transmis dans le bloc de données considérés. Ce système peut être résolu par la méthode de substitution, i.e. résoudre la première équation puis remplacer la solution dans la seconde équation avant de la résoudre. Il est à noter, cependant, que les équations étant polynômiales, les solutions ne sont pas uniques. Après chaque résolution, il faut donc choisir la meilleure solution. Celle-ci est choisie de sorte qu'elle appartienne à l'alphabet de la séquence transmise. Nous allons à présent formaliser la procédure.

Posons :

$$\bar{Y}_b^{(d)}(n) = (y^{(d)}(nK - M - 1) \quad y^{(d)}(nK - M - 2) \quad \dots y^{(d)}(nK - K + M + 1))$$

Appelons $\mathcal{H}_b$ la sous-matrice de $\mathcal{H}$ formée à partir de ses $(L-1)$ dernières lignes, $\bar{W}_b$ le sous-vecteur de $\bar{W}$ formé à partir de ses $(L-1)$ dernières composantes. Posons aussi :

$$\bar{U}_b(n) = (\bar{U}^T(n) \quad \bar{U}^T(n) \circ \bar{U}^T(n) \quad \dots \quad \bar{U}^T(n) \circ \bar{U}^T(n) \dots \circ \bar{U}^T(n))^T$$

où

$$\bar{U}(n) = (u(nL-1) \quad u(nL-2) \quad \dots \quad u(nL-L+1))^T$$

Nous avons donc la relation :

$$\bar{Y}_b(n) = \mathcal{H}_b \bar{U}_b(n) \tag{5.25}$$

qui est un système d'équations polynômiales triangulaires dont les inconnues sont les composantes du vecteur $\bar{U}(n)$. On peut aussi l'écrire comme :

$$\begin{aligned} P_{M+1}(u(nL-1)) &= 0 \\ P_{M+2}(u(nL-1), u(nL-2)) &= 0 \\ &\vdots \\ P_{M+L-1}(u(nL-1), \dots, u(nL-L+1)) &= 0 \end{aligned}$$

Où chaque polynôme P_i , $i = M + 1, \dots, K - M - 1$ est tel que :

$$P_i(u(nL - 1), \dots, u(nL - i + M)) = \sum_{p=1}^P \sum_{j=1}^{i-M} h_p^{(d)}(2M - i + j) u^p(nL - j) - y^{(d)}(nK - i)$$

ou encore

$$P_i(u(nL - 1), \dots, u(nL - i + M)) = \sum_{p=1}^P \sum_{j=1}^{i-M-1} h_p^{(d)}(2M - i + j) u^p(nL - j) - y^{(d)}(nK - i) + \sum_{p=1}^P h_p^{(d)}(M) u^p(nL - i + M)$$

Supposons que les $(i - 1)$ premières équations sont déjà résolues et appelons $\hat{u}(nL - k + M)$, $k = 1, \dots, i - M - 1$, leurs solutions respectives. En remplaçant ces valeurs dans l'expression de $P_i(u(nL - 1), \dots, u(nL - i + M))$, on obtient un polynôme de variable unique $u(nL - i + M)$:

$$\tilde{P}_i(u(nL - i + M)) = \sum_{p=1}^P h_p^{(d)}(M) u^p(nL - i + M) - \tilde{y}^{(d)}(nK - i) \quad (5.26)$$

avec

$$\tilde{y}^{(d)}(nK - i) = y^{(d)}(nK - i) - \sum_{p=1}^P \sum_{j=1}^{i-M-1} h_p^{(d)}(2M - i + j) \hat{u}^p(nL - j) \quad (5.27)$$

Il s'en suit que le symbole recherché $u(nL - i + M)$ est l'une des racines de ce polynôme. Définissons respectivement par $\mathcal{R}_{i,n} = \{r_p\}_{p=1}^P$ l'ensemble des racines de $\tilde{P}_i(u(nL - i + M))$ et $\{\alpha_j\}_{j=1}^Q$ l'alphabet des symboles transmis. Le symbole $\hat{u}(nL - i + M)$ est alors donné par :

$$\hat{u}(nL - i + M) = \arg \min_{r \in \mathcal{R}_{i,n}} \prod_{j=1}^Q (r - \alpha_j) (r - \alpha_j)^* \quad (5.28)$$

La procédure d'estimation des symboles après l'identification du modèle de Hammerstein peut se résumer comme suit :

Algorithme 5.3 Algorithme *TPS-root*³

Pour chaque bloc n , pour $i = M + 1, \dots, K - M - 1$:

1. déterminer $\mathcal{R}_{i,n}$ l'ensemble des racines du polynôme $\tilde{P}_i(u(nL - i + M))$ défini en (5.26) ;

³TPS-root est mis pour Triangular Polynomial System root

2. sélectionner le symbole $\hat{u}(nL - i + M)$ suivant (5.28).

Le principe de l'estimation des symboles par résolution d'une équation polynômiale que suit l'algorithme proposé a aussi été considéré par Redfern et Tong Zhou [115] (voir la section 1.8.4). Cependant, dans leur développement, l'estimation des coefficients des noyaux de Volterra n'est pas considérée, l'hypothèse de la connaissance de ces coefficients étant faite. Dans ce cadre là, i.e. lorsque le canal est connu, il a été montré que l'estimation des symboles par résolution d'équations polynômiales donne de meilleures performances que les autres méthodes d'égalisation des systèmes de Volterra rencontrées dans la littérature à savoir l'inverse d'ordre P [121, 119, 22], la méthode du point fixe [129, 92], les égaliseurs non-linéaires de type Wiener [8] et les égaliseurs DFE non-linéaires [11].

5.3.4.2 Estimation par détermination de l'inverse d'un polynôme

Dans la méthode précédente, l'estimation des symboles se fait en recherchant les racines de (5.26), i.e.

$$\tilde{P}_i(u(nL - i + M)) = 0 \Leftrightarrow \sum_{p=1}^P h_p^{(d)}(M) u^p(nL - i + M) = \hat{y}^{(d)}(nK - i)$$

Nommons $H_M^{(d)}(u(nL - i + M))$ le polynôme en la variable $u(nL - i + M)$ de coefficients $h_p^{(d)}(M)$. Supposons que ce polynôme soit inversible, i.e., il existe un polynôme $J_M^{(d)}$ tel que

$$J_M^{(d)}(\hat{y}^{(d)}(nK - i)) = J_M^{(d)}\left(H_M^{(d)}(u(nL - i + M))\right) = u(nL - i + M) \quad (5.29)$$

Il est à noter qu'en l'absence du bruit $H_M^{(d)}$ caractérise une transformation d'un ensemble fini (l'alphabet de la source) vers un autre ensemble fini. De manière réciproque, $J_M^{(d)}$ est la transformation inverse d'un ensemble fini à un autre. Il est donc toujours possible de déterminer un polynôme de degré fini qui caractérise une telle transformation. Etant donné que l'alphabet auquel appartient la source et les coefficients du polynôme $H_M^{(d)}$ sont connus, nous pouvons recourir à la formule d'interpolation de Lagrange :

Définition 5.1 Soit le polynôme $\mathcal{P}$ de variable x et des couples de mesures (β_j, α_j) , $j = 1, \dots, Q$, telles que $\mathcal{P}(\beta_j) = \alpha_j$. L'expression de ce polynôme est alors donné par :

$$\mathcal{P}(x) = \sum_{j=1}^Q \alpha_j \prod_{k \neq j} \left(\frac{x - \beta_k}{\beta_j - \beta_k} \right) \quad (5.30)$$

Afin d'appliquer cette formule à notre problème, il est nécessaire de déterminer les images des éléments de l'alphabet de la source par le polynôme $H_M^{(d)}$, i.e.

$$\beta_j = H_M^{(d)}(\alpha_j)$$

où les α_j sont les éléments de l'alphabet auquel appartient la source. Connaissant les couples de mesures (β_j, α_j) , $j = 1, \dots, Q$ et appliquant la formule de Lagrange ci-dessus nous pouvons donc déterminer le polynôme inverse $J_M^{(d)}$. Par suite, les symboles estimés sont donnés par :

$$\hat{u}(nL - i + M) = J_M^{(d)}(\tilde{y}^{(d)}(nK - i)), \quad i = M + 1, \dots, K - M - 1 \quad (5.31)$$

ou encore par

$$\hat{u}(nL - i + M) = \sum_{j=1}^Q \alpha_j \prod_{k \neq j} \left(\frac{\tilde{y}^{(d)}(nK - i) - H_M(\alpha_k)}{H_M(\alpha_j) - H_M(\alpha_k)} \right), \quad i = M + 1, \dots, K - M - 1 \quad (5.32)$$

avec $H_M^{(d)}(\alpha_j) = \sum_{p=1}^P h_p^{(d)}(M) \alpha_j^p$

La procédure d'estimation des symboles résumée ci-après sera dénommée, par la suite, **PI** pour (Polynomial Inverse) :

Algorithme 5.4 *Algorithme PI*

1. Déterminer les images $\beta_j = H_M^{(d)}(\alpha_j)$, $j = 1, \dots, Q$ où α_j est un élément de l'alphabet auquel appartient la source.
2. Déterminer le polynôme $J_M^{(d)}$ selon l'équation (5.30).
3. Pour tout n et pour $i = M + 1, \dots, K - M - 1$, déterminer le symbole transmis $u(nL - i + M)$ selon l'équation (5.31).

L'algorithme PI est à différencier de la méthode supervisée d'égalisation des systèmes de Volterra dite "d'inverse d'ordre P ". En effet, dans celle-ci il est question de déterminer un filtre de Volterra tel que sa mise en cascade avec le filtre de Volterra représentant le canal estimé soit équivalent à un filtre de Volterra dont les noyaux d'ordre inférieur à P sont nuls à l'exception du noyau d'ordre 1 représenté par un Dirac. Afin de garantir l'existence de l'inverse d'ordre P , le noyau d'ordre 1 doit être à phase minimale. Dans notre cas, l'existence d'un polynôme inverse est garantie par le fait que le canal réalise une transformation d'un ensemble fini vers un autre ensemble fini. La détermination de l'inverse se base sur les techniques bien connues d'interpolation polynômiale.

Les algorithmes TPS-root et PI s'appliquent aussi bien pour des systèmes ayant un noyau du premier ordre à phase minimale ou non. Cependant, une limitation possible est le fait que ces méthodes soient de type DFE. Une erreur d'estimation peut donc se répercuter sur les symboles suivants. Cependant, le protocole de transmission étant une transmission en blocs complètement découplés, la propagation d'erreur est donc limitée au seul bloc courant.

5.3.4.3 Simulation

Nous considérons de nouveau le système décrit par la relation (5.24). L'entrée est toujours un signal PAM-4. $N = 100$ blocs de taille $L = 4$ sont transmis. La séquence pilote est périodique de période T . Pour différentes périodes de la séquence pilote, le taux d'erreur d'estimation Pe est représenté en fonction du rapport signal à bruit sur les figures 5.8 et 5.9 respectivement pour les algorithmes TPS-root et PI. Ce taux est calculé comme suit :

$$Pe = \frac{\sum_{n=0}^{N-1} \sum_{i=M+1}^{K-M-1} e(nL - i + M)}{N(L-1)}$$

où $e(nL - i)$ est l'erreur de décision définie comme suit :

$$e(nL - i + M) = \begin{cases} 1 & \text{si } u(nL - i + M) \neq \text{Dec}(\hat{u}(nL - i + M)) \\ 0 & \text{si } u(nL - i + M) = \text{Dec}(\hat{u}(nL - i + M)) \end{cases}$$

Dec étant l'élément de décision.

Il est à noter que ces résultats sont la moyenne de 100 expériences indépendantes.

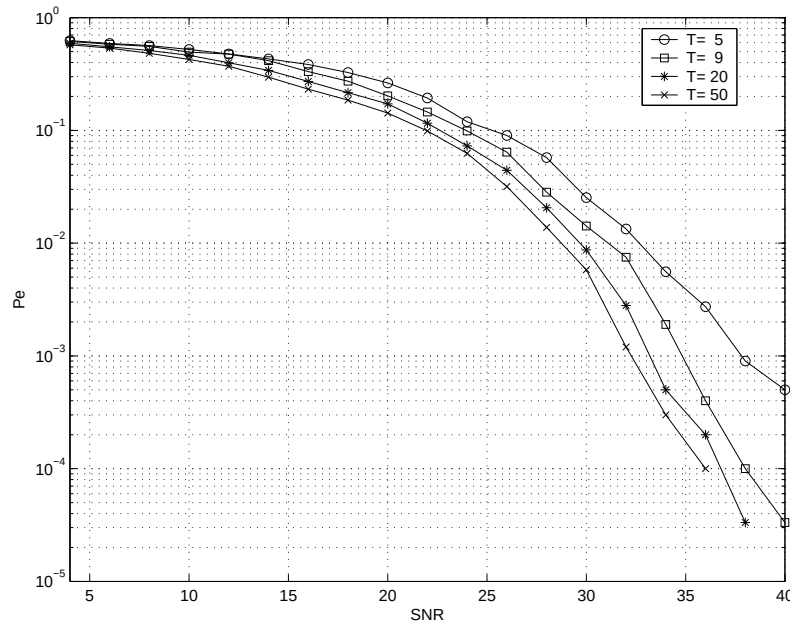


FIG. 5.8 – Taux d'erreur d'estimation des symboles de l'algorithme TPS-root pour différentes périodes de la séquence pilote

Comme on pouvait s'y attendre, les deux algorithmes donnent une très bonne estimation pour des niveaux de bruit faibles. Lorsque le niveau de bruit est fort les deux algorithmes présentent des performances similaires. Mais de manière globale, comme illustré par la figure 5.10, nous pouvons noter que l'algorithme PI donne de meilleures performances que TPS-root. Il est d'ailleurs plus simple à implémenter et utilise moins de ressources de calcul que TPS-root. En effet, pour ce dernier, un système d'équations

5.3. Identification et égalisation semi-aveugle de la liaison descendante

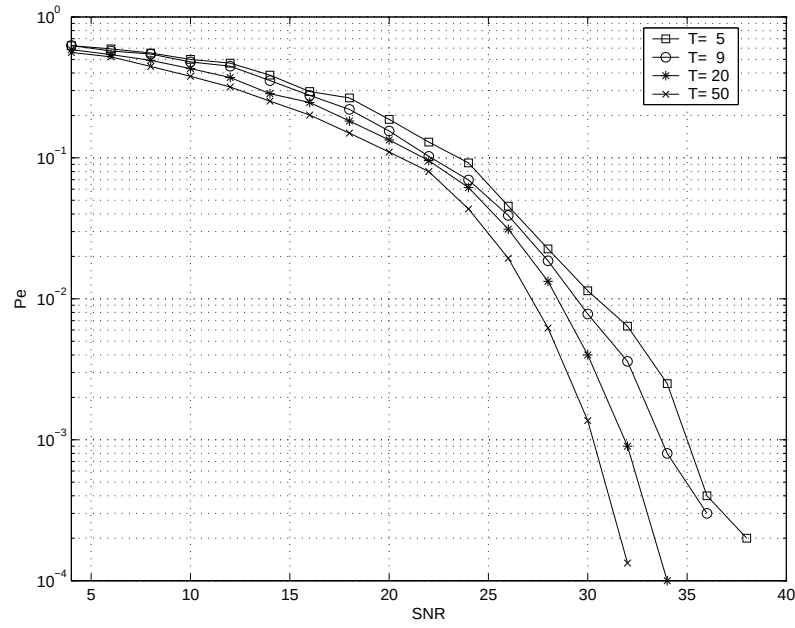


FIG. 5.9 – Taux d'erreur d'estimation des symboles de l'algorithme PI pour différentes périodes de la séquence pilote

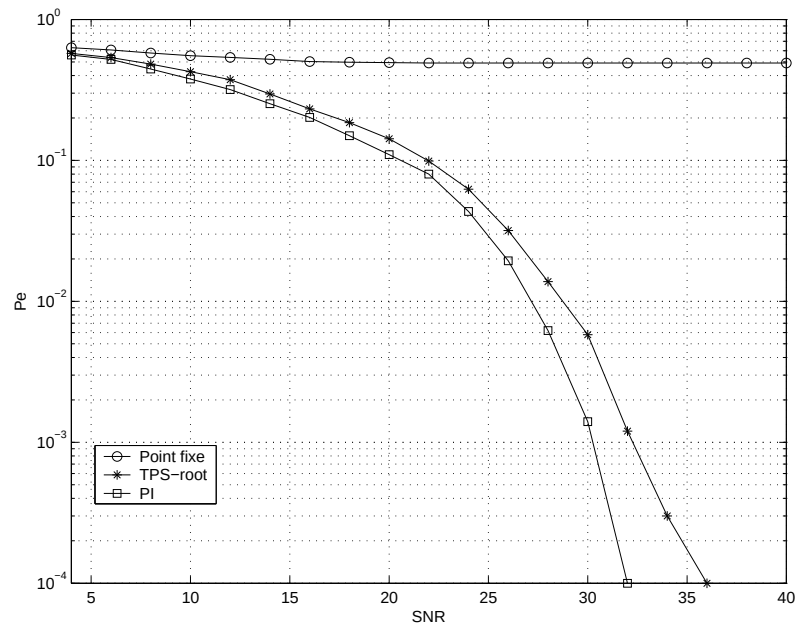


FIG. 5.10 – Comparaison des performances des algorithmes TPS-root, PI et de la méthode du point fixe

est à résoudre pour chaque bloc de données alors que PI réalise l'estimation de l'inverse une fois pour toutes. Nous noterons que dans le cas du système simulé, le taux d'erreur d'estimation pour la méthode du point fixe reste très élevé.

5.4 Identification et égalisation de la liaison ascendante : cas mono-capteur

Dans cette section, nous allons considérer la représentation mono-capteur de la liaison ascendante. Rappelons que celle-ci est représentée par un modèle de Wiener tel que :

$$y^{(a)}(n) = \sum_{p=1}^P \sum_{i_1=0}^M \cdots \sum_{i_p=i_{p-1}}^M h_p^{(a)}(i_1, \dots, i_p) \prod_{j=1}^p u(n - i_j)$$

$$h_p^{(a)}(i_1, \dots, i_p) = \beta_p \prod_{j=1}^p h(i_j), \quad \text{avec } \beta_p = c_p \alpha^p$$

A l'instar du développement réalisé dans la section précédente, nous considérons une transmission en bloc des données et la même procédure de précodage. La représentation vectorielle de la relation d'entrée-sortie du système est donnée par :

$$\begin{aligned} \bar{Y}^{(a)}(n) &= \begin{pmatrix} y^{(a)}(nK) & \cdots & y^{(a)}(nK - K + M + 1) \end{pmatrix}^T \\ &= \mathcal{H}^{(a)} \bar{X}(n) + \bar{W}(n) \end{aligned} \quad (5.33)$$

où

•

$$\mathcal{H}^{(a)} = \begin{pmatrix} \mathcal{H}_1^{(a)} & \cdots & \mathcal{H}_P^{(a)} & \mathcal{H}_{int}^{(a)} \end{pmatrix}$$

Pour $p = 1, \dots, P$:

$$\mathcal{H}_p^{(a)} = \begin{pmatrix} h_p^{(a)}(0, \dots, 0) & \cdots & h_p^{(a)}(M, \dots, M) & \cdots & 0 \\ & \ddots & & \ddots & \\ & & h_p^{(a)}(0, \dots, 0) & \cdots & h_p^{(a)}(M, \dots, M) \end{pmatrix}$$

$\mathcal{H}_{int}^{(a)}$ est une matrice bloc, dont chaque bloc est de type Toeplitz, qui contient les coefficients non-diagonaux des noyaux.

•

$$\bar{X}(n) = \begin{pmatrix} X^T(n) & X^T(n) \circ X^T(n) & \cdots & X^T(n) \circ X^T(n) \cdots \circ X^T(n) \\ X_{int}(n) \end{pmatrix}$$

$X_{int}(n)$ étant le vecteur des produits croisés des composantes de $X(n)$

•

$$\bar{W}(n) = \begin{pmatrix} w(nK) & w(nK-1) & \cdots & w(nK-K+M+1) \end{pmatrix}^T$$

En développant (5.33) tout en tenant compte de (5.10), on peut remarquer que les $(M+1)$ premiers éléments du bloc $\bar{Y}^{(a)}(n)$ sont donnés par une relation similaire à la relation (5.13), i.e. pour $i = 0, \dots, M$:

$$y^{(a)}(nK-i) = \sum_{p=1}^P h_p^{(a)}(M-i, \dots, M-i) u^p(nL) + w(nK-i) \quad (5.34)$$

Quant aux $(L-1)$ éléments restants, ils sont donnés par :

$$\begin{aligned} y^{(d)}(nK-i) &= \sum_{p=1}^P \sum_{j_1=1}^{i-M} \cdots \sum_{j_p=j_{p-1}}^{i-M} h_p^{(a)}(2M-i+j_1, \dots, 2M-i+j_p) \prod_{k=1}^p u(nL-j_k) \\ &\quad + w(nK-i), \quad i = M+1, \dots, K-M-1 \end{aligned} \quad (5.35)$$

La relation (5.34) servant à l'identification du système étant similaire à (5.13) utilisée pour l'identification du modèle de Hammerstein représentant la liaison descendante, les procédures d'estimation qui ont été proposées dans la section 5.3 peuvent donc être appliquées. Cependant, il est à noter que cette relation ne fait intervenir que les coefficients diagonaux des noyaux. La connaissance des termes diagonaux et la propriété de séparabilité des noyaux sont à la base de la procédure d'estimation des paramètres non-diagonaux que nous allons développer. Nous formulons le résultat suivant :

Théorème 5.1 *Les éléments non-diagonaux d'un modèle de Wiener sont obtenus à partir de ses éléments diagonaux, $h_p(i, \dots, i)$, $p = 1, \dots, P$, de la manière suivante :*

$$h_p^{(a)}(i_1, \dots, i_p) = \frac{1}{M+1} \sum_{i=0}^M \frac{h_p^{(a)}(i, \dots, i)}{\left(h_1^{(a)}(i)\right)^p} \prod_{j=1}^p h_1^{(a)}(i_j) \quad (5.36)$$

Preuve : Nous savons qu'en général les coefficients du noyau de Volterra d'ordre p correspondant à un modèle de Wiener sont donnés par (5.4) :

$$h_p^{(a)}(i_1, \dots, i_p) = \beta_p \prod_{j=1}^p h(i_j), \quad p = 1, \dots, P$$

où h est la réponse impulsionnelle du filtre linéaire et les β_p sont liés aux coefficients du polynôme. Les éléments diagonaux sont tels que :

$$h_p^{(a)}(i, \dots, i) = \beta_p (h(i))^p, \quad i = 0; \dots, M$$

Par suite, nous avons :

$$\beta_p = \frac{h_p^{(a)}(i, \dots, i)}{(h(i))^p}, \quad i = 0; \dots, M$$

En sommant cette relation pour toutes les valeurs possibles de i , on obtient :

$$\beta_p = \frac{1}{M+1} \sum_{i=0}^M \frac{h_p^{(a)}(i, \dots, i)}{(h(i))^p}$$

Par ailleurs, sachant que $h_1^{(a)}(i) = \beta_1 h(i)$, i.e. $h(i) = \frac{h_1^{(a)}(i)}{\beta_1}$, nous pouvons écrire que :

$$\beta_p = \frac{\beta_1^p}{M+1} \sum_{i=0}^M \frac{h_p^{(a)}(i, \dots, i)}{(h_1^{(a)}(i))^p}$$

d'où :

$$\frac{\beta_p}{\beta_1^p} = \frac{1}{M+1} \sum_{i=0}^M \frac{h_p^{(a)}(i, \dots, i)}{(h_1^{(a)}(i))^p}$$

Nous avons aussi :

$$\begin{aligned} h_p^{(a)}(i_1, \dots, i_p) &= \beta_p \prod_{j=1}^p h(i_j) \\ &= \beta_p \prod_{j=1}^p \frac{h_1^{(a)}(i_j)}{\beta_1} \\ &= \frac{\beta_p}{\beta_1^p} \prod_{j=1}^p h_1^{(a)}(i_j) \\ &= \frac{1}{M+1} \sum_{i=0}^M \frac{h_p^{(a)}(i, \dots, i)}{(h_1^{(a)}(i))^p} \prod_{j=1}^p h_1^{(a)}(i_j) \end{aligned}$$

ce qui termine la preuve du théorème. ■

Grâce au théorème ci-dessus, la procédure d'identification proposée, permet donc d'estimer tous les paramètres d'un modèle de Wiener. Il est à noter qu' à partir de la seule connaissance des coefficients d'un modèle de Wiener, il n'est pas possible de déterminer d'une part la réponse impulsionnelle du filtre linéaire et les coefficients du polynôme de l'autre sans ambiguïté de signe. Des hypothèses supplémentaires sont alors nécessaires pour lever les ambiguïtés possibles.

En vue de l'estimation du signal informatif, la procédure d'estimation par la résolution d'un système triangulaire d'équations polynômiales peut-être appliquée.

5.4.1 Simulation

En vue d'évaluer la procédure ci-dessus, nous considérons l'identification d'un canal non-linéaire formé par la mise en cascade d'un filtre linéaire RIF dont les coefficients de la réponse impulsionnelle sont $h = (1 \ 0.5 \ -0.2)^T$ et d'un polynôme statique

5.5. Récepteur autodidacte de la liaison ascendante : cas multi-capteurs

donné par $P(u) = u + 0.3u^2 + 0.1u^3$. Le signal informatif est de type PAM-4. Nous nous restreignons au cas où la séquence pilote est périodique. Les coefficients des noyaux de Volterra correspondant au canal ainsi que leurs estimés sont donnés par les tableaux 5.5, 5.6 et 5.7. Ces résultats sont obtenus en moyennant 100 expériences indépendantes.

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_1(0) = 1$	1.001 ± 0.003	0.991 ± 0.034	1.019 ± 0.030
$h_1(1) = 0.5$	0.501 ± 0.001	0.495 ± 0.017	0.504 ± 0.009
$h_1(2) = -0.2$	$-0.200 \pm 6 \times 10^{-4}$	-0.199 ± 0.006	-0.197 ± 0.008

TAB. 5.5 – Estimation des coefficients du noyau d'ordre 1 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_2(0,0) = 0.30$	0.301 ± 0.006	0.275 ± 0.052	0.312 ± 0.127
$h_2(0,1) = 0.15$	0.151 ± 0.003	0.138 ± 0.027	0.152 ± 0.056
$h_2(0,2) = -0.06$	-0.06 ± 0.001	-0.055 ± 0.011	-0.060 ± 0.023
$h_2(1,1) = 0.075$	0.075 ± 0.001	0.069 ± 0.015	0.075 ± 0.027
$h_2(1,2) = -0.03$	$-0.03 \pm 5 \times 10^{-4}$	-0.028 ± 0.006	-0.029 ± 0.011
$h_2(2,2) = 0.012$	$0.012 \pm 2 \times 10^{-4}$	0.011 ± 0.003	0.012 ± 0.004

TAB. 5.6 – Estimation des coefficients du noyau d'ordre 2 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$

Nous pouvons noter qu'en moyenne l'identification du canal se fait correctement. L'écart-type autour de la valeur moyenne est d'autant plus grand que le bruit additif est important. Par ailleurs, nous observons à nouveau, au travers de l'évaluation de l'EQMEN (tableau 5.8), que l'estimation du modèle est d'autant plus précise que la période est élevée.

5.5 Récepteur autodidacte de la liaison ascendante : cas multi-capteurs

Considérons le précodage suivant, suggéré par [79]. Il est tel que la séquence transmise $x(n)$, $n > 0$, provienne de la séquence informative $u(n)$ en insérant M zéros entre chaque symbole :

$$x(n(M+1) - i) = \begin{cases} u(n) & \text{si } i = M \\ 0 & \text{si } i = 0, 1, \dots, M-1 \end{cases} \quad (5.37)$$

Valeurs réelles	Valeurs estimées		
	30 dB	10 dB	5 dB
$h_3(0,0,0) = 0.100$	0.099 ± 0.022	0.124 ± 0.282	0.140 ± 0.626
$h_3(0,0,1) = 0.050$	0.050 ± 0.011	0.062 ± 0.139	0.066 ± 0.310
$h_3(0,0,2) = -0.02$	-0.02 ± 0.004	-0.025 ± 0.055	-0.028 ± 0.122
$h_3(0,1,1) = 0.025$	0.025 ± 0.005	0.031 ± 0.069	0.032 ± 0.155
$h_3(0,1,2) = -0.01$	-0.010 ± 0.002	-0.012 ± 0.027	-0.013 ± 0.061
$h_3(0,2,2) = 0.004$	$0.004 \pm 9 \times 10^{-4}$	0.005 ± 0.011	0.006 ± 0.024
$h_3(1,1,1) = 0.012$	0.012 ± 0.003	0.015 ± 0.034	0.015 ± 0.078
$h_3(1,1,2) = -5 \times 10^{-3}$	-0.005 ± 0.001	-0.006 ± 0.014	-0.006 ± 0.030
$h_3(1,2,2) = 0.002$	$0.002 \pm 3 \times 10^{-4}$	0.002 ± 0.005	0.003 ± 0.012
$h_3(2,2,2) = -8 \times 10^{-4}$	$-8 \times 10^{-4} \pm 2 \times 10^{-4}$	-0.001 ± 0.002	-0.001 ± 0.005

 TAB. 5.7 – Estimation des coefficients du noyau d'ordre 3 d'un modèle de Wiener à l'aide d'une séquence pilote périodique de période $T = 50$

Période T	SNR		
	30 dB	10 dB	5 dB
5	9.600×10^{-3}	1.910	7.826
10	4.400×10^{-3}	0.362	4.511
20	2.400×10^{-3}	0.268	1.968
50	6.594×10^{-4}	0.070	0.105

TAB. 5.8 – Evaluation de l'EQMEN pour différentes périodes de la séquence pilote

Ce précodage réduit les paramètres effectifs du modèle de Volterra aux seuls coefficients diagonaux. En effet, tous les produits croisés de la séquence x pour des décalages temporels différents sont nuls. Rappelons la relation (5.2) :

$$y_k^{(a)}(n(M+1)-i) = \sum_{p=1}^P \sum_{i_1=0}^M \cdots \sum_{i_p=i_{p-1}}^M h_{p,k}^{(a)}(i_1, \dots, i_p) \prod_{j=1}^p x(n(M+1)-i-i_j) + w_k(n(M+1)-i), \quad i = 0, \dots, M$$

Tenant compte de (5.37), on peut remarquer que $x(n(M+1)-i-i_j)$ n'est non nul, et égal à $u(n)$, que pour $i+i_j = M$, i.e. $i_j = M-i$, $j = 1, \dots, P$. Par suite nous avons :

$$y_k^{(a)}(n(M+1)-i) = \sum_{p=1}^P h_{p,k}^{(a)}(M-i, \dots, M-i) u^p(n) + w_k(n(M+1)-i), \quad i = 0, \dots, M \quad (5.38)$$

Les sorties $y_k^{(a)}(n(M+1)-i)$, $i = 0, \dots, M$ dépendent de la même entrée $u(n)$. L'information au niveau du récepteur est donc redondante relativement au signal informatif. Dans [79] un sous-échantillonnage du signal reçu est proposé afin d'éviter cette redondance. L'égalisation est ensuite réalisée en utilisant les statistiques d'ordre deux du signal reçu. Dans notre cas, nous allons plutôt exploiter la redondance introduite comme une diversité supplémentaire. Nous adoptons les notations suivantes :

$$h_{p,k,i} = h_{p,k}^{(a)}(M-i, \dots, M-i)$$

$$y_{k,n,i} = y_k^{(a)}(n(M+1) - i)$$

$$w_{k,n,i} = w_k(n(M+1) - i)$$

D'où :

$$y_{k,n,i} = \sum_{p=1}^P h_{p,k,i} u^p(n) + w_{k,n,i}, \quad \begin{array}{l} k = 0, \dots, K_a - 1 \\ n = 0, \dots, N - 1 \\ i = 0, \dots, M \end{array} \quad (5.39)$$

Où K_a est le nombre de capteurs utilisés à la réception et N le nombre de blocs de mesures. Les mesures $y_{k,n,i}$ peuvent être groupées dans un tenseur d'ordre 3 de même que les coefficients du canal $h_{p,k,i}$ tandis que les $u^p(n)$ peuvent être groupés dans une matrice. Tenant compte de la factorisation (5.1), nous obtenons la relation suivante :

$$y_{k,n,i} = \sum_{p=1}^P \beta_{k,p} h^p(M - i) u^p(n) + w_{k,n,i} \quad (5.40)$$

Nous pouvons reconnaître la représentation scalaire d'un modèle PARAFAC (voir l'annexe A.2). Ainsi, en appliquant une décomposition PARAFAC au tenseur $\mathbb{Y}$ on peut donc récupérer le signal d'entrée, et les paramètres effectifs du modèle de Volterra représentant les canaux de transmission.

Nous définissons les matrices suivantes :

– $\mathbf{Y}_{k..}$ de dimensions $(M+1) \times N$:

$$\mathbf{Y}_{k..} = \begin{pmatrix} y_{k,0,0} & y_{k,1,0} & \cdots & y_{k,N-1,0} \\ y_{k,0,1} & y_{k,1,1} & \cdots & y_{k,N-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{k,0,M} & y_{k,1,M} & \cdots & y_{k,N-1,M} \end{pmatrix}, \quad k = 0, \dots, K_a - 1$$

– $\mathbf{H}$ de dimensions $(M+1) \times P$

$$\mathbf{H} = \begin{pmatrix} h(M) & h^2(M) & \cdots & h^P(M) \\ h(M-1) & h^2(M-1) & \cdots & h^P(M-1) \\ \vdots & \vdots & \ddots & \vdots \\ h(0) & h^2(0) & \cdots & h^P(0) \end{pmatrix}$$

– $\mathbf{U}$ de dimensions $P \times N$

$$\mathbf{U} = \begin{pmatrix} u(0) & u(1) & \cdots & u(N-1) \\ u^2(0) & u^2(1) & \cdots & u^2(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ u^P(0) & u^P(1) & \cdots & u^P(N-1) \end{pmatrix}$$

– $\mathbf{B}$ de dimensions $K_a \times P$

$$\mathbf{B} = \begin{pmatrix} \beta_{0,1} & \beta_{0,2} & \cdots & \beta_{0,P} \\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,P} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{K_a-1,1} & \beta_{K_a-1,2} & \cdots & \beta_{K_a-1,P} \end{pmatrix}$$

Rappelons que les paramètres $\beta_{k,p}$ peuvent s'écrire de deux manières différentes selon qu'un multiplexage est utilisé ou non (voir le paragraphe 5.2.2). Dans le cas sans multiplexage nous avons montré que $\beta_{k,p} = c_{p,k} \alpha_k^p$ tandis qu'avec multiplexage $\beta_{k,p} = c_p \alpha_k^p$. Pour chacun de ces deux cas, la matrice $\mathbf{B}$ peut respectivement s'écrire comme :

$$\mathbf{B} = \begin{pmatrix} c_{1,0} \alpha_0 & c_{2,0} \alpha_0^2 & \cdots & c_{P,0} \alpha_0^P \\ c_{1,1} \alpha_1 & c_{2,1} \alpha_1^2 & \cdots & c_{P,1} \alpha_1^P \\ \vdots & \vdots & \ddots & \vdots \\ c_{1,K_a-1} \alpha_{K_a-1} & c_{2,K_a-1} \alpha_{K_a-1}^2 & \cdots & c_{P,K_a-1} \alpha_{K_a-1}^P \end{pmatrix}$$

et

$$\mathbf{B} = \begin{pmatrix} \alpha_0 & \alpha_0^2 & \cdots & \alpha_0^P \\ \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^P \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{K_a-1} & \alpha_{K_a-1}^2 & \cdots & \alpha_{K_a-1}^P \end{pmatrix} \begin{pmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_P \end{pmatrix}$$

Nous notons que $\mathbf{Y}_{k..}$ admet la factorisation suivante :

$$\mathbf{Y}_{k..} = \mathbf{H} D_k(\mathbf{B}) \mathbf{U}, \quad k = 0, \dots, K_a - 1 \quad (5.41)$$

où $D_k(\cdot)$ est l'opérateur qui construit une matrice diagonale avec la $(k+1)$ -ème ligne de la matrice en argument. De même, nous définissons aussi les matrices $\mathbf{Y}_{.n}$ et $\mathbf{Y}_{..i}$ de dimensions respectives $(K_a \times (M+1))$ et $(N \times K_a)$:

$$\mathbf{Y}_{.n} = \begin{pmatrix} y_{0,n,0} & y_{0,n,1} & \cdots & y_{0,n,M} \\ y_{1,n,0} & y_{1,n,1} & \cdots & y_{1,n,M} \\ \vdots & \vdots & \ddots & \vdots \\ y_{K_a-1,n,0} & y_{K_a-1,n,1} & \cdots & y_{K_a-1,n,M} \end{pmatrix} = \mathbf{B} D_n(\mathbf{U}^T) \mathbf{H}^T, \quad n = 0, \dots, N-1 \quad (5.42)$$

et

$$\mathbf{Y}_{..i} = \begin{pmatrix} y_{0,0,i} & y_{1,0,i} & \cdots & y_{K_a-1,0,i} \\ y_{0,1,i} & y_{1,1,i} & \cdots & y_{K_a-1,1,i} \\ \vdots & \vdots & \ddots & \vdots \\ y_{0,N-1,i} & y_{1,N-1,i} & \cdots & y_{K_a-1,N-1,i} \end{pmatrix} = \mathbf{U}^T D_i(\mathbf{H}) \mathbf{B}^T, \quad i = 0, \dots, M \quad (5.43)$$

Définition 5.2 [73] Considérons la matrice $\mathbf{A} \in \mathbb{C}^{I \times J}$. Si tout ensemble de $l < J$ colonnes de la matrice $\mathbf{A}$ sont linéairement indépendantes et cessent de l'être lorsqu'on ajoute une $(l+1)$ -ème colonne, alors le rang de Kruskal de la matrice $\mathbf{A}$, noté $k_{\mathbf{A}}$, est égal à l .

Remarque 5.4 On a $k_{\mathbf{A}} \leq \text{rang}(\mathbf{A})$

L'unicité de la décomposition PARAFAC est garantie par le théorème suivant :

Théorème 5.2 *Considérons l'ensemble des K_a matrices $\mathbf{Y}_{k..} = \mathbf{H}\mathbf{D}_k(\mathbf{B})\mathbf{U}$, $k = 0, \dots, K_a - 1$, où $\mathbf{H} \in \mathbb{C}^{(M+1) \times P}$, $\mathbf{U} \in \mathbb{C}^{P \times N}$, $\mathbf{B} \in \mathbb{C}^{K_a \times P}$. Si $k_{\mathbf{H}} + k_{\mathbf{B}} + k_{\mathbf{U}^T} \geq 2P + 2$, alors les matrices $\mathbf{H}$, $\mathbf{B}$ et $\mathbf{U}^T$ sont uniques à une matrice de permutation et une matrice diagonale près.*

Ce théorème a d'abord été prouvé dans le cas de données réelles [73] puis étendu au cas de données complexes par [122].

Corollaire 5.1 *Si le signal u prend au moins $P \leq N$ valeurs distinctes non nulles alors la décomposition PARAFAC est unique si $k_{\mathbf{H}} + k_{\mathbf{B}} \geq P + 2$.*

Preuve : Tenant compte de la structure de Vandermonde de la matrice $\mathbf{U}^T$, si u prend $P - 1$ valeurs distinctes, nous n'aurions que $P - 1$ vecteurs lignes distincts. La matrice serait alors de rang déficient. $\mathbf{U}^T$ est donc de rang plein si u prend au moins P valeurs distinctes. Par suite, $k_{\mathbf{U}^T}$, son rang de Kruskal, est donc égal à P , son rang. Ceci permet de modifier la condition nécessaire du théorème 5.2 comme formulé par le corollaire 5.1. ■

Par ailleurs, nous pouvons remarquer que la matrice $\mathbf{H}$ est aussi de type Vandermonde. Nous en déduisons le corollaire suivant :

Corollaire 5.2 *Si le signal u prend au moins $P \leq N$ valeurs distinctes non nulles et si le canal est tel qu'au moins $P \leq M + 1$ de ses coefficients $h(i)$ sont distincts alors la décomposition PARAFAC est unique si $k_{\mathbf{B}} \geq 2$.*

Ce corollaire est particulièrement intéressant dans le cas où un multiplexage est utilisé. En effet la factorisation de la matrice $\mathbf{B}$ en un produit d'une matrice de Vandermonde et d'une matrice diagonale permet de définir une borne au nombre de capteurs nécessaires pour assurer l'unicité de la décomposition PARAFAC. En effet, il suffirait d'avoir $K_a \geq 2$ capteurs, c'est à dire au moins deux capteurs de caractéristiques différentes ($\alpha_0 \neq \alpha_1$).

Il est à noter que l'unicité est entendue, à une matrice de permutation et un facteur d'échelle près. En d'autres termes, tout autre ensemble de matrices $\bar{\mathbf{H}}$, $\bar{\mathbf{B}}$ et $\bar{\mathbf{U}}^T$ pouvant reconstituer les matrices $\mathbf{Y}_{k..}$, $k = 0, \dots, K_a - 1$ est relié à l'ensemble de matrices $\mathbf{H}$, $\mathbf{B}$ et $\mathbf{U}^T$ par :

$$\bar{\mathbf{H}} = \mathbf{H}\mathbf{\Pi}\mathbf{\Delta}_1, \quad \bar{\mathbf{B}} = \mathbf{B}\mathbf{\Pi}\mathbf{\Delta}_2, \quad \bar{\mathbf{U}}^T = \mathbf{U}^T\mathbf{\Pi}\mathbf{\Delta}_3$$

où $\mathbf{\Pi}$ est une matrice de permutation et les $\mathbf{\Delta}_i$, $i = 1, 2, 3$ sont des matrices diagonales telles que

$$\mathbf{\Delta}_1\mathbf{\Delta}_2\mathbf{\Delta}_3 = \mathbf{I}$$

Il est à noter que P étant fini, l'ensemble des matrices de permutation est aussi fini. Cependant, il est nécessaire de trouver un moyen pour lever l'ambiguïté sur les solutions. Nous mettons à profit la structure des matrices pour lever l'ambiguïté due à la

permutation des colonnes. Rappelons que la matrice $\mathbf{U}^T$ est telle que :

$$\mathbf{U}^T = \begin{pmatrix} u(0) & u^2(0) & \cdots & u^P(0) \\ u(1) & u^2(1) & \cdots & u^P(1) \\ \vdots & \vdots & \ddots & \vdots \\ u(N-1) & u^2(N-1) & \cdots & u^P(N-1) \end{pmatrix}$$

Nous définissons la mesure suivante :

$$\rho_p = \frac{\max |\mathbf{U}_{\cdot p}^T|}{\min |\mathbf{U}_{\cdot p}^T|} \quad (5.44)$$

où $\max |\mathbf{U}_{\cdot p}^T|$ et $\min |\mathbf{U}_{\cdot p}^T|$ sont respectivement le plus grand et le plus petit des modules des éléments de la p -ème colonne de $\mathbf{U}^T$. Cette mesure croît strictement avec l'ordre de la non-linéarité. Elle est invariante par rapport à tout facteur d'échelle affectant une colonne donnée. En conséquence, une fois la matrice $\mathbf{U}^T$ estimée, on calcule ρ_p correspondant à chacune des colonnes puis on les ordonne dans un ordre croissant de ρ_p . Cette opération permet de définir la matrice de permutation à appliquer à $\mathbf{U}^T$ de sorte que ses colonnes soient à ρ_p croissant. Nous avons ainsi levé toute ambiguïté liée à la permutation des colonnes. En somme, la décomposition PARAFAC sous les contraintes définies par le théorème 5.2 et la contrainte "*colonne de $\mathbf{U}^T$ à ρ_p croissant*" est unique à un facteur d'échelle près. En d'autres termes, tout autre ensemble de matrices $\bar{\mathbf{H}}$, $\bar{\mathbf{B}}$ et $\bar{\mathbf{U}}^T$ pouvant reconstituer les matrices $\mathbf{Y}_{k..}$, $k = 0, \dots, K_a - 1$ est relié à l'ensemble de matrices $\mathbf{H}$, $\mathbf{B}$ et $\mathbf{U}^T$ par :

$$\bar{\mathbf{H}} = \mathbf{H}\Delta_1, \quad \bar{\mathbf{B}} = \mathbf{B}\Delta_2, \quad \bar{\mathbf{U}}^T = \mathbf{U}^T\Delta_3$$

avec $\Delta_1\Delta_2\Delta_3 = \mathbf{I}$. Notons par ailleurs qu'en utilisant un correcteur de gain, l'effet du facteur d'échelle sur la matrice $\mathbf{U}^T$ peut être complètement levé. L'indétermination ne subsiste donc que sur l'estimation du canal.

5.5.1 Algorithme ALS pour l'estimation des paramètres

Le principe de l'algorithme ALS est le suivant : à chaque étape, mettre à jour un sous ensemble de paramètres en utilisant l'algorithme des moindres carrés conditionné par les estimés antérieurs des autres sous ensembles de paramètres. Puis, refaire le processus de manière alternée, i.e. pour un autre sous ensemble de paramètres jusqu'à convergence. Les matrices $\mathbf{Y}_{k..}$ étant données, leur concaténation permet d'écrire :

$$\begin{pmatrix} \mathbf{Y}_{0..} \\ \vdots \\ \mathbf{Y}_{K_a-1..} \end{pmatrix} = \begin{pmatrix} \mathbf{H}\mathbf{D}_0(\mathbf{B}) \\ \vdots \\ \mathbf{H}\mathbf{D}_{K_a-1}(\mathbf{B}) \end{pmatrix} \mathbf{U} \quad (5.45)$$

En notant $\tilde{\mathbf{Y}}_{k..}$ la version bruitée de $\mathbf{Y}_{k..}$, et en considérant les versions précédemment estimées de $\mathbf{H}$ et de $\mathbf{B}$, que nous notons $\hat{\mathbf{H}}$ et $\hat{\mathbf{B}}$, le problème des moindres carrés est

alors formulé comme suit :

$$\min_{\mathbf{U}} \left\| \begin{pmatrix} \tilde{\mathbf{Y}}_{0..} \\ \vdots \\ \tilde{\mathbf{Y}}_{K_a-1..} \end{pmatrix} - \begin{pmatrix} \hat{\mathbf{H}}D_0(\hat{\mathbf{B}}) \\ \vdots \\ \hat{\mathbf{H}}D_{K_a-1}(\hat{\mathbf{B}}) \end{pmatrix} \mathbf{U} \right\|_F \quad (5.46)$$

où $\|\cdot\|_F$ est la norme de Frobenius. La solution à ce problème est donnée par :

$$\hat{\mathbf{U}} = \begin{pmatrix} \hat{\mathbf{H}}D_0(\hat{\mathbf{B}}) \\ \vdots \\ \hat{\mathbf{H}}D_{K_a-1}(\hat{\mathbf{B}}) \end{pmatrix}^\dagger \begin{pmatrix} \tilde{\mathbf{Y}}_{0..} \\ \vdots \\ \tilde{\mathbf{Y}}_{K_a-1..} \end{pmatrix} \quad (5.47)$$

où $\dagger$ indique la pseudo-inverse.

De même, sachant que $\mathbf{Y}_{.n} = \mathbf{B}D_n(\mathbf{U}^T)\mathbf{H}^T$, $n = 0, \dots, N-1$ et $\mathbf{Y}_{..i} = \mathbf{U}^T D_i(\mathbf{H})\mathbf{B}^T$, $i = 0, \dots, M$, les estimés au sens des moindres carrés des matrices $\mathbf{B}$ et $\mathbf{H}$ sont données par :

$$\hat{\mathbf{H}}^T = \begin{pmatrix} \hat{\mathbf{B}}D_0(\hat{\mathbf{U}}^T) \\ \vdots \\ \hat{\mathbf{B}}D_{N-1}(\hat{\mathbf{U}}^T) \end{pmatrix}^\dagger \begin{pmatrix} \tilde{\mathbf{Y}}_{.0.} \\ \vdots \\ \tilde{\mathbf{Y}}_{.N-1.} \end{pmatrix} \quad (5.48)$$

$$\hat{\mathbf{B}}^T = \begin{pmatrix} \hat{\mathbf{U}}^T D_0(\hat{\mathbf{H}}) \\ \vdots \\ \hat{\mathbf{U}}^T D_M(\hat{\mathbf{H}}) \end{pmatrix}^\dagger \begin{pmatrix} \tilde{\mathbf{Y}}_{..0} \\ \vdots \\ \tilde{\mathbf{Y}}_{..M} \end{pmatrix} \quad (5.49)$$

Il est à noter que les matrices à estimer sont initialisées avec des valeurs aléatoires. La complexité de l'algorithme correspond au coût de trois pseudo-inversions à chaque itération.

Remarque 5.5 *Plusieurs méthodes d'égalisation des systèmes SIMO non-linéaires existent dans la littérature [45, 113, 78, 79]. Nous ne pouvons pas les comparer à la présente procédure car elles requièrent que les différents canaux n'aient aucun facteur commun, i.e. les canaux ne doivent avoir aucun zéro commun, ce qui n'est pas le cas de la modélisation envisagée ici.*

5.5.2 Simulations

Nous considérons le cas où un multiplexeur est utilisé. Le polynôme représentant la non-linéarité statique est le même que celui considéré dans le cas monocapteur, soit $P(u) = u + 0.3u^2 + 0.1u^3$. La réponse impulsionnelle est définie telle que $h = (1 \ 0.5 \ -0.2)^T$, $M = 2$. Chaque capteur est caractérisé par un gain différent. La séquence d'information est de type PAM-4. Pour un nombre de données $N = 500$, nous traçons le taux d'erreur Pe pour différentes valeurs de K_a , le nombre de capteurs. L'évaluation des performances est de type Monte-Carlo avec 100 expériences indépendantes.

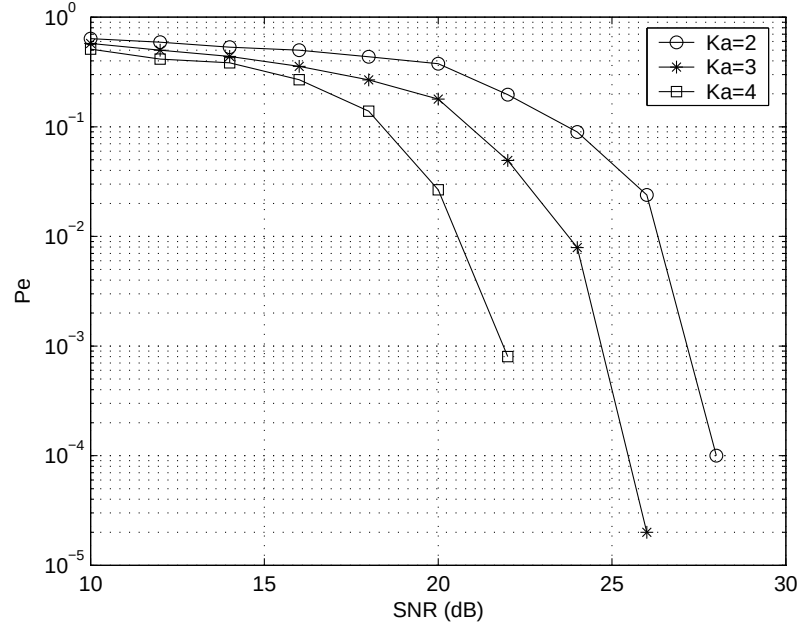


FIG. 5.11 – Taux d'erreur d'estimation des symboles en fonction du niveau de bruit (N=500)

Nous pouvons noter que le taux d'erreur est d'autant plus petit que le nombre de capteurs s'accroît. En environnement fortement bruité, la diversité introduite par les capteurs n'améliore que faiblement le taux d'erreur. Cette amélioration est plutôt marquée lorsque le niveau de bruit diminue. Avec 4 capteurs, au delà de 22dB il n'y a plus d'erreur alors qu'avec 2 capteurs le taux d'erreur ne s'annule que pour un SNR supérieur à 28dB.

5.6 Récepteur hybride TPS-root PARAFAC de la liaison ascendante

Le grand écueil de l'approche PARAFAC est de recourir à un précodeur qui réduit fortement le taux de transmission. Nous pouvons toutefois coupler cette procédure avec la méthode TPS-root que nous avons proposée dans les sections précédentes. Considérons de nouveau le protocole de transmission en bloc considéré dans les sections 3 et 4.

Considérons un bloc d'entrées

$$U(n) = \begin{pmatrix} u(nL) & u(nL-1) & \cdots & u(nL-L+1) \end{pmatrix}^T \quad (5.50)$$

où L est la taille du bloc. En utilisant la matrice de redondance $\mathbf{C}$, de dimensions $K \times L$

($K > L$), définie par les équations (5.7) et (5.8), le bloc de données suivant est généré

$$X(n) = \left(\underbrace{0 \cdots 0}_M u(nL) \underbrace{0 \cdots 0}_M u(nL-1) \cdots u(nL-L+1) \right)^T$$

Le signal reçu par chaque capteur

$$\bar{Y}_k^{(a)}(n) = \left(y_k^{(a)}(nK) \ y_k^{(a)}(nK-1) \ \cdots \ y_k^{(a)}(nK-K+M+1) \right)^T, \quad k=0, \dots, K_a-1$$

a ses $(M+1)$ premiers éléments donnés par la relation suivante déduite de (5.34) :

$$y_k^{(a)}(nK-i) = \sum_{p=1}^P h_{p,k}^{(a)}(M-i, \dots, M-i) u^p(nL) + w(nK-i) \quad (5.51)$$

Quant aux $(L-1)$ derniers éléments, ils sont le fruit d'un mélange convolutif et non-linéaire des $(L-1)$ dernières composantes du bloc $U(n)$:

$$\begin{aligned} y_k^{(a)}(nK-i) &= \sum_{p=1}^P \sum_{j_1=1}^{i-M} \cdots \sum_{j_p=j_{p-1}}^{i-M} h_{p,k}^{(a)}(2M-i+j_1, \dots, 2M-i+j_p) \prod_{k=1}^p u(nL-j_k) \\ &\quad + w(nK-i), \quad i = M+1, \dots, K-M-1 \end{aligned} \quad (5.52)$$

Tenant compte de la factorisation (5.1), la relation (5.51) peut être réécrite comme suit :

$$y_k^{(a)}(nK-i) = \sum_{p=1}^P \beta_{k,p} h^p(M-i) u^p(nL) + w_k(nK-i) \quad (5.53)$$

En considérant les $y_{k,n,i} = y_k^{(a)}(nK-i)$ comme étant les éléments d'un tenseur d'ordre trois, nous pouvons reconnaître en (5.53) la représentation scalaire d'un modèle PARAFAC. Ainsi, en appliquant une décomposition PARAFAC au tenseur d'éléments $y_{k,n,i}$ on peut donc récupérer le signal d'entrée, et les coefficients diagonaux du modèle de Wiener correspondant à chaque canal.

En reprenant les mêmes définitions des matrices $\mathbf{Y}_{k..}$, $\mathbf{H}$, $\mathbf{U}$ et $\mathbf{B}$ que dans la section précédente, à l'instar des développements effectués dans la section précédente, nous pouvons noter que $\mathbf{Y}_{k..}$ admet la factorisation suivante :

$$\mathbf{Y}_{k..} = \mathbf{H} \mathbf{D}_k(\mathbf{B}) \mathbf{U}, \quad k=0, \dots, K_a-1 \quad (5.54)$$

De même, nous pouvons aussi écrire

$$\begin{aligned} \mathbf{Y}_{..n} &= \begin{pmatrix} y_0^{(a)}(nK) & y_0^{(a)}(nK-1) & \cdots & y_0^{(a)}(nK-M) \\ y_1^{(a)}(nK) & y_1^{(a)}(nK-1) & \cdots & y_1^{(a)}(nK-M) \\ \vdots & \vdots & \ddots & \vdots \\ y_{K_a-1}^{(a)}(nK) & y_{K_a-1}^{(a)}(nK-1) & \cdots & y_{K_a-1}^{(a)}(nK-M) \end{pmatrix} \\ &= \mathbf{B} \mathbf{D}_n(\mathbf{U}^T) \mathbf{H}^T, \quad n=0, \dots, N-1 \end{aligned} \quad (5.55)$$

et

$$\begin{aligned} \mathbf{Y}_{..i} &= \begin{pmatrix} y_0^{(a)}(-i) & y_1^{(a)}(-i) & \cdots & y_{K_a-1}^{(a)}(-i) \\ y_0^{(a)}(K-i) & y_1^{(a)}(K-i) & \cdots & y_{K_a-1}^{(a)}(K-i) \\ \vdots & \vdots & \ddots & \vdots \\ y_0^{(a)}((N-1)K-i) & y_1^{(a)}((N-1)K-i) & \cdots & y_{K_a-1}^{(a)}((N-1)K-i) \end{pmatrix} \\ &= \mathbf{U}^T D_i(\mathbf{H}) \mathbf{B}^T, \quad i = 0, \dots, M \end{aligned} \quad (5.56)$$

$\mathbf{Y}_{..n}$ et $\mathbf{Y}_{..i}$ étant respectivement de dimensions $K_a \times (M+1)$ et $N \times K_a$.

L'unicité de la décomposition PARAFAC est aussi garantie par le corollaire 5.1. Comme indiqué dans la section précédente, l'unicité est entendue à une permutation des colonnes et un facteur d'échelle près. En utilisant la mesure définie par (5.44) nous pouvons lever l'ambiguïté sur la permutation des colonnes. En conséquence, il est possible d'estimer les symboles $u(nL)$, $n = 0, \dots, N-1$ en utilisant l'algorithme ALS. Il est certes vrai que les coefficients diagonaux du modèle de Wiener représentant les canaux de transmission peuvent aussi être estimés par ce biais mais il demeure une ambiguïté (des facteurs d'échelles) sur l'estimation des $\beta_{k,p}$ et des $h(i)$, qui sont nécessaires à la reconstruction de ces coefficients diagonaux. La séquence $u(nL)$ ayant été estimée, nous pouvons l'utiliser pour estimer les coefficients diagonaux des canaux via la méthode classique des moindres carrés. Ensuite, ces coefficients serviront à déterminer les coefficients non-diagonaux en appliquant le résultat (5.36).

En notant par $\hat{u}(nL)$ les symboles estimés via la décomposition PARAFAC, nous pouvons réécrire (5.51) comme suit :

$$y_k^{(a)}(nK-i) = \sum_{p=1}^P h_{p,k}^{(a)}(M-i, \dots, M-i) \hat{u}^p(nL) + w_k(nK-i), \quad i = 0, \dots, M \quad (5.57)$$

On reconnaît là une équation polynômiale. Le problème est à présent de déterminer les coefficients $h_{p,k}^{(a)}(M-i, \dots, M-i)$ du polynôme. C'est là un problème d'interpolation polynômiale dont la formulation au sens des moindres carrés est :

$$\hat{H}_{k,i} = \arg \min_{H_{k,i}} \|Z_{k,i} - \hat{\mathbf{U}}^T H_{k,i}\|^2 \quad (5.58)$$

où $H_{k,i}$ est le vecteur de dimensions $P \times 1$ contenant les coefficients $h_{p,k}^{(a)}(M-i, \dots, M-i)$, $p = 1, \dots, P$ et $Z_{k,i}$ est la $(i+1)$ -ème colonne de la matrice $\mathbf{Y}_{k..}^T$. La solution au problème (5.58) est donnée en résolvant le système d'équation triangulaire suivant :

$$\mathbf{R} \hat{H}_{k,i} = \mathbf{Q}^T Z_{k,i} \quad (5.59)$$

où $\mathbf{R}$ est une matrice de dimensions $P \times P$ triangulaire supérieure et $\mathbf{Q}$ une matrice orthogonale de dimensions $N \times P$ telles que $\mathbf{U}^T = \mathbf{Q}\mathbf{R}$. Par cette procédure nous pouvons donc déterminer les coefficients diagonaux de chaque canal. La détermination des coefficients non-diagonaux se fait en appliquant la formule (5.36). Une fois le modèle d'un ou de plusieurs canaux déterminés, les symboles émis non encore estimés sont alors déterminés en appliquant la procédure TPS-root.

Algorithme 5.5 Algorithme d'identification et d'égalisation aveugle PARAFAC-TPS-root

1. Etant données les mesures du signal reçu $y_k^{(a)}(nK - i)$, $i = 0, \dots, M$, $k = 0, \dots, K_a - 1$, $n = 0, \dots, N - 1$, constituer les matrices $\mathbf{Y}_{k..}$, $\mathbf{Y}_{..n}$ et $\mathbf{Y}_{..i}$.
2. Appliquer l'algorithme ALS pour l'estimation de la matrice $\mathbf{U}^T$ en appliquant la procédure nécessaire pour lever toute ambiguïté.
3. Faire la décomposition QR de la matrice estimée $\mathbf{U}^T$
4. Déterminer les coefficients diagonaux des canaux en résolvant (5.59) pour $i = 0, \dots, M$ et $k = 0, \dots, K_a - 1$
5. Connaissant les éléments diagonaux des canaux, appliquer la formule (5.36) pour estimer les coefficients non-diagonaux des canaux.
6. Le modèle de chaque canal étant identifié, estimer les symboles restant en appliquant la procédure TPS-root.

Il est à noter que la procédure TPS-root a été définie dans le cas SISO. La connaissance du modèle d'un canal suffit donc pour appliquer la procédure TPS-root. Etant donné que nous avons déterminé les modèles de tous les canaux de la représentation multi-capteur, nous pouvons donc constituer des systèmes d'équations polynômiales sur-déterminés pour l'estimation des symboles. Nous considérons l'équation (5.52) que nous réécrivons ci-après dans le cas sans bruit et après un changement de variable :

$$y_k^{(a)}(nK - M - i) = \sum_{p=1}^P \sum_{j_1=1}^i \cdots \sum_{j_p=j_{p-1}}^i h_{p,k}^{(a)}(M - i + j_1, \dots, M - i + j_p) \prod_{m=1}^p u(nL - j_m)$$

avec $i = 1, \dots, L - 1$. En supposant que les symboles $u(nL - j)$, $j = 1, \dots, i - 1$, ont déjà été estimés, nous pouvons écrire le terme homogène d'ordre p de la manière suivante :

$$\begin{aligned} & \sum_{j_1=1}^i \cdots \sum_{j_p=j_{p-1}}^i h_{p,k}^{(a)}(M - i + j_1, \dots, M - i + j_p) \prod_{m=1}^p u(nL - j_m) \\ &= \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=j_{p-1}}^{i-1} h_{p,k}^{(a)}(M - i + j_1, \dots, M - i + j_p) \prod_{m=1}^p u(nL - j_m) \\ & \quad + u(nL - i) \sum_{j_1=1}^{i-1} \cdots \sum_{j_{p-1}=j_{p-2}}^{i-1} h_{p,k}^{(a)}(M - i + j_1, \dots, M - i + j_{p-1}, M) \prod_{m=1}^{p-1} u(nL - j_m) \\ & \quad + u^2(nL - i) \sum_{j_1=1}^{i-1} \cdots \sum_{j_{p-2}=j_{p-3}}^{i-1} h_{p,k}^{(a)}(M - i + j_1, \dots, M - i + j_{p-2}, M, M) \prod_{m=1}^{p-2} u(nL - j_m) \\ & \quad \vdots \\ & \quad + u^{p-1}(nL - i) \sum_{j_1=1}^{i-1} h_{p,k}^{(a)}(M - i + j_1, M, \dots, M) u(nL - j_1) \\ & \quad + h_{p,k}^{(a)}(M, \dots, M) u^p(nL - i) \\ &= \sum_{q=0}^p \gamma_{q,p,k,n,i} u^q(nL - i) \end{aligned}$$

avec

$$\gamma_{q,p,k,n,i} = \sum_{j_1=1}^{i-1} \cdots \sum_{j_{p-q}=j_{p-q-1}}^{i-1} h_{p,k}^{(a)}(M-i+j_1, \dots, M-i+j_{p-q}, \underbrace{M, \dots, M}_{q \text{ termes}}) \prod_{m=1}^{p-q} u(nL-j_m) \quad (5.60)$$

Par suite, nous avons :

$$y_k^{(a)}(nK-M-i) = \sum_{p=1}^P \sum_{q=0}^p \gamma_{q,p,k,n,i} u^q(nL-i) = \sum_{q=0}^P \bar{\gamma}_{q,k,n,i} u^q(nL-i) \quad (5.61)$$

où

$$\bar{\gamma}_{q,k,n,i} = \sum_{p=q}^P \gamma_{q,p,k,n,i}, \quad \gamma_{0,0,k,n,i} = 0 \quad (5.62)$$

En posant :

$$Y_{n,i} = \begin{pmatrix} y_0^{(a)}(nK-M-i) - \bar{\gamma}_{0,0,n,i} \\ y_1^{(a)}(nK-M-i) - \bar{\gamma}_{0,1,n,i} \\ \vdots \\ y_{K_a-1}^{(a)}(nK-M-i) - \bar{\gamma}_{0,K_a-1,n,i} \end{pmatrix}, \quad U_{n,i} = \begin{pmatrix} u^P(nL-i) \\ u^{P-1}(nL-i) \\ \vdots \\ u(nL-i) \end{pmatrix}$$

et

$$\mathbf{\Lambda}_{n,i} = \begin{pmatrix} \bar{\gamma}_{P,0,n,i} & \bar{\gamma}_{P-1,0,n,i} & \cdots & \bar{\gamma}_{1,0,n,i} \\ \bar{\gamma}_{P,1,n,i} & \bar{\gamma}_{P-1,1,n,i} & \cdots & \bar{\gamma}_{1,1,n,i} \\ \vdots & \vdots & \cdots & \vdots \\ \bar{\gamma}_{P,K_a-1,n,i} & \bar{\gamma}_{P-1,K_a-1,n,i} & \cdots & \bar{\gamma}_{1,K_a-1,n,i} \end{pmatrix}$$

nous obtenons l'équation suivante

$$Y_{n,i} = \mathbf{\Lambda}_{n,i} U_{n,i} \quad (5.63)$$

La matrice $\mathbf{\Lambda}_{n,i}$ est de dimensions $K_a \times P$. En supposant que $K_a \geq P$, nous pouvons réaliser la factorisation QR de la matrice $\mathbf{\Lambda}_{n,i}$, c'est à dire $\mathbf{\Lambda}_{n,i} = \mathbf{Q}_{n,i} \mathbf{R}_{n,i}$, où $\mathbf{Q}_{n,i}$ est une matrice orthogonale de dimensions $K_a \times P$ et $\mathbf{R}_{n,i}$ est une matrice triangulaire supérieure de dimensions $P \times P$. On a alors :

$$\mathbf{Q}_{n,i}^T Y_{n,i} = \mathbf{R}_{n,i} U_{n,i}$$

Tenant compte de la structure triangulaire supérieure de la matrice $\mathbf{R}_{n,i}$, nous pouvons remarquer que $u(nL-i)$, le symbole à estimer, est égal, à un facteur d'échelle près, à la dernière composante du vecteur $\mathbf{Q}_{n,i}^T Y_{n,i}$. En notant par $(Q_{n,i})_{.M}$ la dernière colonne de la matrice $\mathbf{Q}_{n,i}$ et $R_{n,i}(P, P)$ l'élément situé sur la P -ème ligne et la P -ème colonne de la matrice $\mathbf{R}_{n,i}$, on a :

$$(Q_{n,i})_{.M}^T Y_{n,i} = R_{n,i}(P, P) u(nL-i)$$

d'où :

$$u(nL-i) = \frac{(Q_{n,i})_{.M}^T Y_{n,i}}{R_{n,i}(P, P)} \quad (5.64)$$

Cette nouvelle procédure d'estimation aveugle conjointe des canaux et des symboles est résumée ci-après :

Algorithme 5.6 *Algorithme d'identification et d'égalisation aveugle PARAFAC-TPS-QR*

1. Etant données les mesures du signal reçu $y_k^{(a)}(nK - i)$, $i = 0, \dots, M$, $k = 0, \dots, K_a - 1$, $n = 0, \dots, N - 1$, constituer les matrices $\mathbf{Y}_{k..}$, $\mathbf{Y}_{..n}$ et $\mathbf{Y}_{..i}$.
 - (a) Appliquer l'algorithme ALS pour l'estimation de la matrice $\mathbf{U}^T$ en appliquant la procédure nécessaire pour lever toute ambiguïté.
 - (b) Faire la décomposition QR de la matrice estimée $\mathbf{U}^T$
 - (c) Déterminer les coefficients diagonaux des canaux en résolvant (5.59) pour $i = 0, \dots, M$ et $k = 0, \dots, K_a - 1$
 - (d) Connaissant les éléments diagonaux des canaux, appliquer la formule (5.36) pour estimer les coefficients non-diagonaux des canaux.
2. Le modèle de chaque canal étant identifié, estimer les symboles restant en appliquant la procédure TPS-QR suivante :
 - (a) pour $n = 0, \dots, N - 1$, $i = 1, \dots, L - 1$, former le vecteur $Y_{n,i}$, déterminer la matrice $\mathbf{\Lambda}_{n,i}$ et faire sa décomposition QR : $\mathbf{\Lambda}_{n,i} = \mathbf{Q}_{n,i}\mathbf{R}_{n,i}$.
 - (b) Déterminer le symbole $u(nL - i)$ en appliquant l'équation (5.64).

5.7 Conclusion

Dans la présente étude, différents algorithmes d'identification et d'égalisation ont été proposés. Les méthodes sont purement déterministes et utilisent un précodage de l'entrée. Ce précodage introduit une certaine redondance au niveau du signal reçu. Celle-ci est exploitée pour recouvrir tous les paramètres du canal qu'il soit représenté par un modèle de Hammerstein ou par un modèle de Wiener. L'extension au cas plus général du modèle de Volterra nécessite un précodage qui réduit davantage le taux d'information transmis. Nous avons aussi considéré le cas multicanal (SIMO). Dans ce cas précis, nous avons montré que le canal de transmission peut avoir une représentation tensorielle, la redondance, l'espace et le temps étant les trois diversités introduites. Une décomposition de type PARAFAC peut être appliquée au tenseur formé par le signal reçu. Nous avons aussi introduit une procédure qui permet de ramener l'ambiguïté des estimés du modèle PARAFAC à un facteur d'échelle près. Par cette décomposition, le signal émis, le canal et le gain de chaque antenne sont conjointement estimés. Le défaut de cette méthode vient du précodage utilisé. Pour palier à cette limitation, une autre méthode a été proposée en se basant sur la décomposition PARAFAC.

Conclusion générale et perspectives

Le travail présenté dans cette thèse a porté sur l'étude des systèmes non-linéaires représentés à l'aide d'un modèle de Volterra. Trois axes majeurs ont été abordés :

- La réduction de la complexité paramétrique d'un modèle de Volterra.
- La réduction de la complexité des méthodes d'estimation paramétrique d'un modèle de Volterra, au sens de l'erreur quadratique moyenne minimale.
- L'identification et l'égalisation semi-aveugle et aveugle de systèmes représentés par un modèle de Volterra simplifié de type Hammerstein ou Wiener.

Nous avons montré dans le chapitre 3 que la complexité paramétrique d'un modèle de Volterra peut être réduite en utilisant un développement de ses noyaux sur des bases de fonctions orthogonales. Nous avons particulièrement étudié les représentations sur des bases de Laguerre et sur les bases plus générales de type BFOR. La parcimonie du développement des noyaux de Volterra sur une base orthogonale étant liée au choix de la base, nous avons développé de nouveaux outils d'optimisation tant dans le cas d'une base de Laguerre que dans celui d'une BFOR.

Dans le cas de la base de Laguerre, des algorithmes d'optimisation de pôle basés sur la seule estimation des coefficients de Fourier ont été présentés. Ces algorithmes ont pour fondement la détermination d'une solution analytique au problème d'optimisation du pôle de Laguerre. Dans le cas d'une BFOR, le problème est plus complexe. Nous avons apporté trois types de solution. La première est basée sur une recherche exhaustive des pôles dans un ensemble prédéterminé. L'avantage de cette approche est que nous pouvons tenir compte de la connaissance *a priori* que nous avons sur le système. La recherche des pôles étant exhaustive, nous ne sommes pas soumis au problème des minima locaux. Toutefois, la méthode peut s'avérer coûteuse en temps de calcul. Nous avons alors présenté une alternative qui consiste à remplacer la recherche exhaustive par la recherche itérative du pôle optimal de Laguerre pour différents sous-systèmes. Nous avons appelé cette approche, approche analytique. Enfin, une troisième approche a été proposée. Elle combine les approches de décomposition matricielle et la représentation sur les bases orthogonales. Dans le cas d'un noyau quadratique séparable, nous avons montré comment une réduction de complexité drastique pouvait être réalisée et avons présenté un algorithme adaptatif.

Dans le chapitre 4, nous avons présenté des méthodes orthogonales d'identification sous deux angles principaux. Dans un premier temps nous avons exploité les propriétés

statistiques des entrées de type PAM et QAM pour générer des bases de polynômes orthogonaux multivariables. En développant le modèle de Volterra sur une telle base, nous avons montré que l'estimation paramétrique du modèle peut se faire sans aucune inversion matricielle. Nous avons aussi exploité le formalisme des bases de polynômes orthogonaux dans le cas de l'estimation adaptative. Il en résulte une amélioration de la vitesse de convergence de l'algorithme LMS sans un surcoût de calcul. Nous avons ensuite mis en place un module d'orthogonalisation des entrées filtrées par la BFOR. Nous avons ainsi couplé la réduction de complexité du modèle et la simplification des procédures d'estimation. Enfin, nous avons montré que dans le cas d'une entrée gaussienne le couplage du formalisme des polynômes orthogonaux et des bases de fonctions orthonormales rationnelles permet de simplifier davantage l'estimation paramétrique tant en bloc que de manière adaptative.

Dans le chapitre 5, nous avons développé de nouveaux outils d'identification et d'égalisation déterministes. En considérant un système de communication concret, nous avons montré que grâce à l'utilisation d'un certain précodage de la séquence informative il est possible d'estimer conjointement un canal non-linéaire, représenté par des cas particuliers de modèle de Volterra tels que le modèle de Hammerstein et le modèle de Wiener, et le signal informatif. Une extension au cas multi-capteurs a aussi été considérée. Pour une telle configuration, grâce à un précodage spécifique de la séquence d'entrée, nous exploitons la diversité spatiale introduite par les capteurs et la diversité temporelle de sorte à obtenir une représentation tensorielle du signal reçu. Une décomposition tensorielle de type PARAFAC a alors été appliquée pour l'estimation conjointe des canaux et de la séquence émise.

Au-delà des contributions de cette thèse, certaines questions demeurent ouvertes et nécessitent une investigation approfondie. Tout d'abord, les propriétés de convergence de l'algorithme adaptatif de l'estimation du pôle de Laguerre sont à établir de manière plus formelle. De même, la détermination analytique des pôles optimaux d'une BFOR de Ninness et Gustafsson est aussi à explorer, d'abord dans le cas des pôles réels puis dans le cas plus général incluant des pôles réels et complexes.

Pour évaluer la complexité paramétrique d'un modèle de Volterra deux paramètres sont à considérer : sa mémoire et son ordre de non-linéarité. L'approche de réduction de complexité paramétrique à l'aide de bases de fonctions orthogonales rationnelles n'opère que sur le facteur mémoire. Est-il possible de construire une nouvelle famille de fonctions orthogonales influant aussi bien sur la mémoire que sur l'ordre de non-linéarité ? Il nous semble que c'est un point intéressant qui mériterait d'être étudié. Aussi, la réduction de la complexité paramétrique n'est pas une fin en soi, encore faut-il que les procédures d'estimation soient simplifiées. Dans le cas gaussien, nous avons montré qu'il est possible de coupler les aspects réduction de complexité paramétrique et simplification des procédures d'estimation via le formalisme des polynômes orthogonaux et le développement sur une BFOR. Est-il possible de réaliser une telle configuration pour des entrées non-gaussiennes ?

Une autre perspective intéressante est d'appliquer les résultats obtenus dans le chapitre 3 à l'annulation d'échos acoustiques. En effet comme expliqué dans le chapitre 1, le modèle de propagation d'échos est non-linéaire avec une mémoire pouvant être très importante. Quant aux résultats du chapitre 4, une application au problème d'annulation de bruit peut être envisagée.

S'agissant de l'utilisation des techniques de décomposition tensorielle pour l'estimation des symboles, à notre connaissance, c'est là une approche toute nouvelle dans le cas des modèles non-linéaires. L'évaluation de la méthode par des simulations est à approfondir et son extension au cas plus général du modèle de Volterra à noyaux non-séparables à investiguer. Par ailleurs, toutes les méthodes d'identification et d'égalisation présentées dans le chapitre 5 sont déterministes et généralement peu robustes au bruit. Il est donc intéressant de "robustifier" ces méthodes par rapport au bruit.

Annexes

Annexe A

Produit de Kronecker et décomposition PARAFAC

A.1 Le produit de Kronecker

Etant donné deux matrices A et B de dimensions respectives $(m \times n)$ et $(p \times q)$, on appelle produit de Kronecker de A par B la matrice partitionnée de dimension $(mp \times nq)$ définie comme :

$$C = A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{pmatrix}$$

En particulier, dans le cas de deux vecteurs a et b de dimension respective m et p , on a :

$$c = a \otimes b = \begin{pmatrix} a_1b \\ a_2b \\ \vdots \\ a_mb \end{pmatrix}$$

c est un vecteur de dimension mp dont les éléments sont de la forme a_ib_j , i.e. $c_{k(i,j)} = a_ib_j$, où :

$$k(i, j) = (i-1)p + j, \quad i = 1, 2, \dots, m; \quad j = 1, 2, \dots, p$$

A.2 La décomposition PARAFAC

Ce modèle trilinéaire a été proposé par Harshman [56] et par Carroll et Chang [23] qui l'ont respectivement appelé PARAFAC (PARallel FACtor analysis) et CANDECOMP (CANonical DECOMPosition). Elle consiste à représenter un tenseur à l'aide de matrices.

La représentation PARAFAC d'un tenseur $\mathbb{H}$ d'ordre trois et de dimensions $N_1 \times N_2 \times N_3$ est définie par la donnée de trois matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{C}$ de dimensions respectives $N_1 \times P$, $N_2 \times P$ et $N_3 \times P$ et d'un tenseur unité cubique $\mathbb{I}$ de dimensions $P \times P \times P$ telle qu'illustrée par la figure A.1.

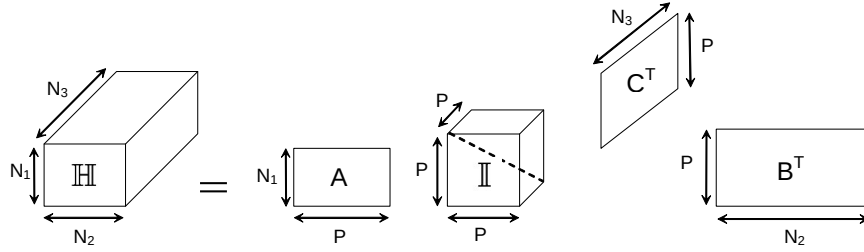


FIG. A.1 – Décomposition PARAFAC

On définit les matrices suivantes :

- $\mathbf{H}_{n_1..}$, $n_1 = 1, \dots, N_1$, de dimensions $(N_2 \times N_3)$ telles que l'élément $\mathbf{H}_{n_1..}(n_2, n_3)$ corresponde à l'élément $\mathbb{H}(n_1, n_2, n_3)$ du tenseur $\mathbb{H}$.
- $\mathbf{H}_{..n_2}$, $n_2 = 1, \dots, N_2$, de dimensions $(N_3 \times N_1)$ telles que l'élément $\mathbf{H}_{..n_2}(n_3, n_1)$ corresponde à l'élément $\mathbb{H}(n_1, n_2, n_3)$ du tenseur $\mathbb{H}$.
- $\mathbf{H}_{..n_3}$, $n_3 = 1, \dots, N_3$, de dimensions $(N_1 \times N_2)$ telles que l'élément $\mathbf{H}_{..n_3}(n_1, n_2)$ corresponde à l'élément $\mathbb{H}(n_1, n_2, n_3)$ du tenseur $\mathbb{H}$.

La construction des matrices $\mathbf{H}_{n_1..}$, $\mathbf{H}_{..n_2}$ et $\mathbf{H}_{..n_3}$ est décrite par la figure A.2.

Chaque élément $\mathbb{H}(n_1, n_2, n_3)$ du tenseur $\mathbb{H}$ peut être écrit en fonction des éléments des matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{C}$ comme décrit par l'équation (A.1). C'est la représentation scalaire de la décomposition PARAFAC :

$$\mathbb{H}(n_1, n_2, n_3) = \sum_{p=1}^P a_{n_1 p} b_{n_2 p} c_{n_3 p} ; \quad \begin{array}{l} n_1 = 1, \dots, N_1 \\ n_2 = 1, \dots, N_2 \\ n_3 = 1, \dots, N_3 \end{array} \quad (\text{A.1})$$

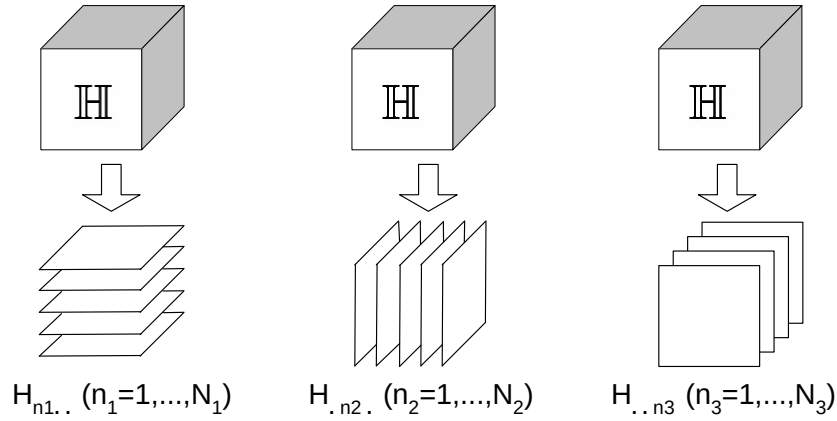


FIG. A.2 – Matrices horizontales, verticales et latérales $\mathbf{H}_{n_1..}$, $\mathbf{H}_{.n_2.}$ et $\mathbf{H}_{..n_3}$

où a_{n_1p} , b_{n_2p} and c_{n_3p} sont respectivement des éléments des matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{C}$.

De cette représentation scalaire, on déduit trois représentations matricielles, chacune étant respectivement associée aux matrices horizontales, verticales et latérales $\mathbf{H}_{n_1..}$, $\mathbf{H}_{.n_2.}$ et $\mathbf{H}_{..n_3}$:

$$\mathbf{H}_{n_1..} = \mathbf{B}D_{n_1}(\mathbf{A})\mathbf{C}^T \quad (\text{A.2})$$

$$\mathbf{H}_{.n_2.} = \mathbf{C}D_{n_2}(\mathbf{B})\mathbf{A}^T \quad (\text{A.3})$$

$$\mathbf{H}_{..n_3} = \mathbf{A}D_{n_3}(\mathbf{C})\mathbf{B}^T \quad (\text{A.4})$$

où $D_k(.)$ est une matrice diagonale dont les éléments diagonaux sont les éléments de la k -ème ligne de la matrice en argument.

Annexe B

Preuves du chapitre 3

B.1 Preuve de la proposition 3.1

A partir de la relation (3.11) et de la définition (3.9), on a :

$$pQ_{1,p} \|h_p\|^2 = \sum_{l=1}^p \sum_{i_l=0}^{\infty} \cdots \sum_{i_p=0}^{\infty} i_l h_p^2(i_1, \dots, i_p)$$

Tenant compte de la relation (3.1), on obtient :

$$\begin{aligned} pQ_{1,p} \|h_p\|^2 &= \sum_{l=1}^p \sum_{i_l=0}^{\infty} \cdots \sum_{i_p=0}^{\infty} \\ &\quad \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} \sum_{q_1=0}^{\infty} \cdots \sum_{q_p=0}^{\infty} i_l g_{k_1, \dots, k_p} g_{q_1, \dots, q_p} \prod_{j=1}^p b_{k_j, p}(i_j) b_{q_j, p}(i_j) \end{aligned}$$

Grâce à la relation (3.6), à la propriété d'orthogonalité de la base et tenant compte de l'expression de $Q_{2,p}$, le développement de ce calcul donne :

$$\begin{aligned} pQ_{1,p} \|h_p\|^2 &= \frac{1 - \xi_p^2}{1 + \xi_p^2} \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (k_1 + \cdots + k_p) g_{k_1, \dots, k_p}^2 - \frac{p \xi_p^2 \|h_p\|^2}{1 + \xi_p^2} \\ &\quad + \frac{2p \xi_p Q_{2,p} \|h_p\|^2}{1 + \xi_p^2} \end{aligned}$$

Par suite :

$$Q_{1,p} = \frac{1 - \xi_p^2}{1 + \xi_p^2} J_p - \frac{\xi_p^2}{1 + \xi_p^2} + \frac{2 \xi_p Q_{2,p}}{1 + \xi_p^2}$$

d'où le résultat (3.12).

B.2 Preuve de la proposition 3.2

Le pôle optimal au sens du critère (3.8) est donné par la résolution du problème suivant :

$$\frac{\partial J_p(\xi_p)}{\partial \xi_p} = 0 \Leftrightarrow Q_{2,p}\xi_p^2 - (2Q_{1,p} + 1)\xi_p + Q_{2,p} = 0$$

En utilisant la définition (3.13), le pôle optimal est donc la racine réelle de l'équation suivante :

$$\xi_p^2 - 2\rho_{0,p}\xi_p + 1 = 0$$

Les racines réelles de cette équation sont obtenues si $|\rho_{0,p}| > 1$. Auquel cas, on a :

$$\xi_{p,opt} = \rho_{0,p} \pm \sqrt{\rho_{0,p}^2 - 1}$$

B.3 Preuve du lemme 3.1

En effet¹ :

$$\begin{aligned} \frac{\partial T_{1,l}}{\partial \xi_p} &= 2 \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) g_{k_1, \dots, k_p} \frac{\partial g_{k_1, \dots, k_p}}{\partial \xi_p} \\ &= \frac{2}{1 - \xi_p^2} \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) g_{k_1, \dots, k_p} \\ &\quad \times \sum_{j=1}^p ((k_j + 1) g_{k_1, \dots, k_j+1, \dots, k_p} - k_j g_{k_1, \dots, k_j-1, \dots, k_p}) \end{aligned}$$

¹Dans ce développement $\delta_{i,j}$ représente le symbole de Kronecker

D'où

$$\begin{aligned}
\frac{1 - \xi_p^2}{2} \frac{\partial T_{1,l}}{\partial \xi_p} &= \sum_{j=1}^p \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_j + 1) g_{k_1, \dots, k_j+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{j=1}^p \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) k_j g_{k_1, \dots, k_j-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&= \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_l + 1) g_{k_1, \dots, k_l+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad + \sum_{j=1}^p (1 - \delta_{j,l}) \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_j + 1) g_{k_1, \dots, k_j+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{j=1}^p (1 - \delta_{j,l}) \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) k_j g_{k_1, \dots, k_j-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&= \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_l + 1) g_{k_1, \dots, k_l+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad + \sum_{j=1}^p (1 - \delta_{j,l}) \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_j + 1) g_{k_1, \dots, k_j+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{j=1}^p (1 - \delta_{j,l}) \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_j + 1) g_{k_1, \dots, k_j, \dots, k_p} g_{k_1, \dots, k_j+1, \dots, k_p} \\
&= \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_l + 1) g_{k_1, \dots, k_l+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} 2k_l^2 g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
&= \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) (k_l + 1) g_{k_1, \dots, k_l+1, \dots, k_p} g_{k_1, \dots, k_p} \\
&\quad - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} 2(k_l + 1)^2 g_{k_1, \dots, k_l, \dots, k_p} g_{k_1, \dots, k_l+1, \dots, k_p}
\end{aligned}$$

$$\begin{aligned}
 & - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
 = & - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (k_l + 1) g_{k_1, \dots, k_l+1, \dots, k_p} g_{k_1, \dots, k_p} \\
 & - \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
 = & -2 \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_l-1, \dots, k_p} g_{k_1, \dots, k_p} \\
 \frac{\partial T_{1,l}}{\partial \xi_p} = & \frac{-2}{1 - \xi_p^2} T_{2,l}
 \end{aligned}$$

De la même manière on montre que

$$\frac{\partial T_{2,l}}{\partial \xi_p} = \frac{-2}{1 - \xi_p^2} T_{1,l}$$

Par suite :

$$\begin{aligned}
 \frac{\partial R_{1,p}}{\partial \xi_p} &= \sum_{l=1}^p \frac{\partial T_{1,l}}{\partial \xi_p} \\
 &= \frac{-2}{1 - \xi_p^2} \sum_{l=1}^p T_{2,l} \\
 &= \frac{-2}{1 - \xi_p^2} R_{2,p}
 \end{aligned}$$

Et de même, on a :

$$\frac{\partial R_{2,p}}{\partial \xi_p} = \frac{-2}{1 - \xi_p^2} R_{1,p}$$

B.4 Preuve de la proposition 3.3

Le moment $R_{1,p}$ peut s'exprimer en fonction de la fonction de coût (3.8). En effet :

$$\begin{aligned}
 R_{1,p} &= \sum_{l=1}^p T_{1,l} = \sum_{l=1}^p \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (2k_l + 1) g_{k_1, \dots, k_p}^2 \\
 &= 2 \sum_{l=1}^p \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} k_l g_{k_1, \dots, k_p}^2 + p \|h_p\|^2 \\
 &= 2 \sum_{k_1=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} (k_1 + \cdots + k_p) g_{k_1, \dots, k_p}^2 + p \|h_p\|^2 \\
 &= 2p \|h_p\|^2 J_p + p \|h_p\|^2
 \end{aligned}$$

d'où

$$\frac{\partial R_{1,p}}{\partial \xi_p} = 2p \|h_p\|^2 \frac{\partial J_p}{\partial \xi_p}$$

Sachant que

$$\frac{\partial J_p}{\partial \xi_p} = \frac{-2}{(1 - \xi_p^2)^2} ((1 + \xi_p^2) Q_{2,p} - 2\xi_p Q_{1,p} - \xi_p)$$

nous obtenons

$$\frac{\partial R_{1,p}}{\partial \xi_p} = \frac{-4p \|h_p\|^2}{(1 - \xi_p^2)^2} ((1 + \xi_p^2) Q_{2,p} - 2\xi_p Q_{1,p} - \xi_p)$$

En utilisant la relation (3.18) du lemme 3.1, nous avons l'un des résultats recherchés, soit :

$$R_{2,p} = \frac{2p \|h_p\|^2}{1 - \xi_p^2} ((1 + \xi_p^2) Q_{2,p} - 2\xi_p Q_{1,p} - \xi_p)$$

C'est la relation (3.21) de la proposition 3.3. Par ailleurs,

$$\frac{\partial R_{2,p}}{\partial \xi_p} = 2p \|h_p\|^2 \frac{4\xi_p Q_{2,p} - 2(1 + \xi_p^2) Q_{1,p} - (1 + \xi_p^2)}{(1 - \xi_p^2)^2}$$

en utilisant (3.19) on obtient alors la relation (3.20).

Annexe C

Polynômes orthogonaux

C.1 Polynômes orthogonaux pour des signaux PAM

C.1.1 Construction des polynômes orthogonaux monovariés pour une variable de type PAM

A partir de la base $1, x, x^2, x^3$, construisons une base orthonormale.

$$P_0(x) = 1$$

$$P_1(x) = x - \langle x, P_0(x) \rangle P_0(x)$$

$$\langle x, P_0(x) \rangle = \mathbb{E}[x] = 0$$

d'où :

$$P_1(x) = x$$

Normalisons à présent ce polynôme. Il va de soi que $|P_1(x)|^2 = \mathbb{E}[x^2] = \mu_2$. Par suite :

$$P_1(x) = \frac{x}{\sqrt{\mu_2}}$$

$$P_2(x) = x^2 - \langle x^2, P_0(x) \rangle P_0(x) - \langle x^2, P_1(x) \rangle P_1(x)$$

avec

$$\langle x^2, P_0(x) \rangle = \mathbb{E}[x^2] = \mu_2$$

$$\langle x^2, P_1(x) \rangle = \frac{1}{\sqrt{\mu_2}} \mathbb{E}[x^3] = 0$$

d'où :

$$P_2(x) = x^2 - \mu_2$$

Sa norme est :

$$|P_2(x)|^2 = \mathbb{E}[(x^2 - \mu_2)^2] = \mu_4 - \mu_2^2$$

Par suite :

$$P_2(x) = \frac{1}{\sqrt{\mu_4 - \mu_2^2}} (x^2 - \mu_2)$$

$$P_3(x) = x^3 - \langle x^3, P_0(x) \rangle P_0(x) - \langle x^3, P_1(x) \rangle P_1(x) - \langle x^3, P_2(x) \rangle P_2(x)$$

avec

$$\langle x^3, P_0(x) \rangle = \mathbb{E}[x^3] = 0$$

$$\langle x^3, P_1(x) \rangle = \frac{1}{\sqrt{\mu_2}} \mathbb{E}[x^4] = \frac{\mu_4}{\sqrt{\mu_2}}$$

$$\langle x^3, P_2(x) \rangle = \frac{1}{\sqrt{\mu_4 - \mu_2^2}} \mathbb{E}[x^5 - \mu_2 x^3] = 0$$

d'où :

$$P_3(x) = x^3 - \frac{\mu_4}{\mu_2} x$$

Sa norme est :

$$|P_3(x)|^2 = \mathbb{E}\left[\left(x^3 - \frac{\mu_4}{\mu_2} x\right)^2\right] = \frac{\mu_2^2 \mu_6 - \mu_4^2 \mu_2}{\mu_2^2}$$

Par suite :

$$P_3(x) = \frac{1}{\sqrt{\mu_2^2 \mu_6 - \mu_4^2 \mu_2}} (\mu_2 x^3 - \mu_4 x)$$

C.1.2 Liens avec les coefficients des noyaux du modèle de Volterra

Selon (4.31),

$$f_i(n_1, \dots, n_i) = \sum_{j=i}^3 \mathbb{E}[V_j(U(n)) Q_{n_1, \dots, n_i}^{(i)}(U(n))]$$

où

$$V_j(U(n)) = \sum_{n_1=0}^{M-1} \cdots \sum_{n_j=n_{j-1}}^{M-1} h_j(n_1, \dots, n_j) u(n - n_1) \cdots u(n - n_j)$$

On a :

$$\begin{aligned} f_3(n_1, n_2, n_3) &= \mathbb{E}[V_3(U(n)) Q_{n_1, n_2, n_3}^{(3)}(U(n))] \\ &= \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=j}^{M-1} h_3(i, j, k) \mathbb{E}[u(n-i)u(n-j)u(n-k) Q_{n_1, n_2, n_3}^{(3)}(U(n))] \end{aligned}$$

Tenant compte de la définition des polynômes $Q_{n_1, n_2, n_3}^{(3)}(U(n))$, on a :

$$\mathbb{E}[u(n-i)u(n-j)u(n-k) Q_{n_1, n_1, n_1}^{(3)}(U(n))] = \begin{cases} \kappa_{3,3} & \text{si } n_1 = i = j = k \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1,n_2,n_2}^{(3)}(U(n))] = \begin{cases} \kappa_{2,2}\kappa_{1,1} & \text{si } \begin{cases} n_1 = i \neq n_2 = j = k \\ n_1 = j \neq n_2 = i = k \\ n_1 = k \neq n_2 = i = j \end{cases} \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1,n_1,n_3}^{(3)}(U(n))] = \begin{cases} \kappa_{2,2}\kappa_{1,1} & \text{si } \begin{cases} n_3 = i \neq n_1 = j = k \\ n_3 = j \neq n_1 = i = k \\ n_3 = k \neq n_1 = i = j \end{cases} \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1,n_2,n_3}^{(3)}(U(n))] = \begin{cases} \kappa_{1,1}^3 & \text{si } \begin{cases} n_1 = i \neq n_2 = j \neq n_3 = k \\ n_1 = i \neq n_2 = k \neq n_3 = j \\ n_1 = j \neq n_2 = i \neq n_3 = k \\ n_1 = j \neq n_2 = k \neq n_3 = i \\ n_1 = k \neq n_2 = i \neq n_3 = j \\ n_1 = k \neq n_2 = j \neq n_3 = i \end{cases} \\ 0 & \text{sinon} \end{cases}$$

d'où l'équation (4.32). On a aussi :

$$\begin{aligned} f_2(n_1, n_2) &= \mathbb{E}[V_2(U(n))Q_{n_1,n_2}^{(2)}(U(n))] + \mathbb{E}[V_3(U(n))Q_{n_1,n_2}^{(2)}(U(n))] \\ &= \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} h_2(i, j) \mathbb{E}[u(n-i)u(n-j)Q_{n_1,n_2}^{(2)}(U(n))] \\ &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=j}^{M-1} h_3(i, j, k) \mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1,n_2}^{(2)}(U(n))] \end{aligned}$$

or $\mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1,n_2}^{(2)}(U(n))]$ est nul car étant une fonction de moments d'ordre 5 de u . Donc :

$$f_2(n_1, n_2) = \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} h_2(i, j) \mathbb{E}[u(n-i)u(n-j)Q_{n_1,n_2}^{(2)}(U(n))]$$

Tenant compte de la définition des polynômes $Q_{n_1,n_2}^{(2)}(U(n))$, on a :

$$\mathbb{E}[u(n-i)u(n-j)Q_{n_1,n_2}^{(2)}(U(n))] = \begin{cases} \kappa_{2,2}, & \text{si } n_1 = n_2 = i = j \\ \kappa_{1,1}^2, & \text{si } n_1 = i \neq n_2 = j \\ \kappa_{1,1}^2, & \text{si } n_1 = j \neq n_2 = i \\ 0 & \text{sinon} \end{cases}$$

d'où l'équation (4.33).

Enfin, pour les termes du premier ordre on a :

$$\begin{aligned}
 f_1(n_1) &= \mathbb{E}[V_1(U(n))Q_{n_1}^{(1)}(U(n))] + \mathbb{E}[V_2(U(n))Q_{n_1}^{(1)}(U(n))] \\
 &\quad + \mathbb{E}[V_3(U(n))Q_{n_1}^{(1)}(U(n))] \\
 &= \sum_{i=0}^{M-1} h_1(i) \mathbb{E}[u(n-i)Q_{n_1}^{(1)}(U(n))] \\
 &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} h_2(i, j) \mathbb{E}[u(n-i)u(n-j)Q_{n_1}^{(1)}(U(n))] \\
 &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=j}^{M-1} h_3(i, j, k) \mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1}^{(1)}(U(n))]
 \end{aligned}$$

Le terme $\mathbb{E}[u(n-i)u(n-j)Q_{n_1}^{(1)}(U(n))]$ est nul car il ne fait intervenir que des moments d'ordre trois de l'entrée. Par suite :

$$\begin{aligned}
 f_1(n_1) &= \sum_{i=0}^{M-1} h_1(i) \mathbb{E}[u(n-i)Q_{n_1}^{(1)}(U(n))] \\
 &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=j}^{M-1} h_3(i, j, k) \mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1}^{(1)}(U(n))]
 \end{aligned}$$

avec :

$$\begin{aligned}
 \mathbb{E}[u(n-i)Q_{n_1}^{(1)}(U(n))] &= \begin{cases} \kappa_{1,1} & \text{si } i = n_1 \\ 0 & \text{sinon} \end{cases} \\
 \mathbb{E}[u(n-i)u(n-j)u(n-k)Q_{n_1}^{(1)}(U(n))] &= \begin{cases} \kappa_{3,1} & \text{si } n_1 = i = j = k \\ \mu_2 \kappa_{1,1} & \text{si } i = j \neq n_1 = k \\ \mu_2 \kappa_{1,1} & \text{si } j = k \neq n_1 = i \\ 0 & \text{sinon} \end{cases}
 \end{aligned}$$

C.2 Polynômes orthogonaux pour des signaux QAM

C.2.1 Construction des polynomes orthogonaux monovariabiles pour une variable de type QAM

A partir de la base $1, x, x^*, x^2, xx^*, x^2x^*$, construisons une base orthonormale.

$$P_{0,0}(x) = 1$$

$$P_{1,0}(x) = x - \langle x, P_{0,0}(x) \rangle P_{0,0}(x)$$

$$\langle x, P_{0,0}(x) \rangle = \mathbb{E}[x] = 0$$

d'où :

$$P_{1,0}(x) = x$$

Normalisons à présent ce polynôme. Il va de soi que $|P_{1,0}(x)|^2 = \mathbb{E}[xx^*] = \mu_{1,1}$. Par suite :

$$P_{1,0}(x) = \frac{x}{\sqrt{\mu_{1,1}}}$$

$$P_{0,1}(x) = x^* - \langle x^*, P_{0,0}(x) \rangle P_{0,0}(x) - \langle x^*, P_{1,0}(x) \rangle P_{1,0}(x)$$

$$\langle x^*, P_{0,0}(x) \rangle = \mathbb{E}[x^*] = 0$$

$$\langle x^*, P_{1,0}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E}[x^* x^*] = 0$$

d'où :

$$P_{0,1}(x) = x^*$$

Par suite de normalisation nous obtenons :

$$P_{0,1}(x) = \frac{x^*}{\sqrt{\mu_{1,1}}}$$

$$P_{2,0}(x) = x^2 - \langle x^2, P_{0,0}(x) \rangle P_{0,0}(x) - \langle x^2, P_{0,1}(x) \rangle P_{0,1}(x) - \langle x^2, P_{1,0}(x) \rangle P_{1,0}(x)$$

avec

$$\langle x^2, P_{0,0}(x) \rangle = \mathbb{E}[x^2] = 0$$

$$\langle x^2, P_{1,0}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E}[x^2 x^*] = 0$$

$$\langle x^2, P_{0,1}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E}[x^3] = 0$$

d'où :

$$P_{2,0}(x) = x^2$$

Sa norme est :

$$|P_{2,0}(x)|^2 = \mathbb{E}[x^2 x^{*2}] = \mu_{2,2}$$

Par suite :

$$P_{2,0}(x) = \frac{x^2}{\sqrt{\mu_{2,2}}}$$

$$\begin{aligned} P_{1,1}(x) &= xx^* - \langle xx^*, P_{0,0}(x) \rangle P_{0,0}(x) - \langle xx^*, P_{0,1}(x) \rangle P_{0,1}(x) \\ &\quad - \langle xx^*, P_{1,0}(x) \rangle P_{1,0}(x) - \langle xx^*, P_{2,0}(x) \rangle P_{2,0}(x) \end{aligned}$$

avec

$$\langle xx^*, P_{0,0}(x) \rangle = \mathbb{E}[xx^*] = \mu_{1,1}$$

$$\langle xx^*, P_{1,0}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E}[xx^* x^*] = 0$$

$$\langle xx^*, P_{0,1}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E}[xx^* x] = 0$$

$$\langle xx^*, P_{2,0}(x) \rangle = \frac{1}{\sqrt{\mu_{2,2}}} \mathbb{E} [xx^*x^{*2}] = 0$$

d'où :

$$P_{1,1}(x) = xx^* - \mu_{1,1}$$

Sa norme est :

$$|P_{1,1}(x)|^2 = \mu_{2,2} - \mu_{1,1}^2$$

Par suite :

$$P_{1,1}(x) = \frac{xx^* - \mu_{1,1}}{\sqrt{\mu_{2,2} - \mu_{1,1}^2}}$$

$$\begin{aligned} P_{2,1}(x) &= x^2x^* - \langle x^2x^*, P_{0,0}(x) \rangle P_{0,0}(x) \\ &\quad - \langle x^2x^*, P_{1,0}(x) \rangle P_{1,0}(x) - \langle x^2x^*, P_{0,1}(x) \rangle P_{0,1}(x) \\ &\quad - \langle x^2x^*, P_{2,0}(x) \rangle P_{2,0}(x) - \langle x^2x^*, P_{1,1}(x) \rangle P_{1,1}(x) \end{aligned}$$

avec

$$\langle x^2x^*, P_{0,0}(x) \rangle = \mathbb{E} [x^2x^*] = 0$$

$$\langle x^2x^*, P_{1,0}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E} [x^2x^*x^*] = \frac{\mu_{2,2}}{\sqrt{\mu_{1,1}}}$$

$$\langle x^2x^*, P_{0,1}(x) \rangle = \frac{1}{\sqrt{\mu_{1,1}}} \mathbb{E} [x^2x^*x] = 0$$

$$\langle x^2x^*, P_{2,0}(x) \rangle = \frac{1}{\sqrt{\mu_{2,2}}} \mathbb{E} [x^2x^*x^{*2}] = 0$$

$$\langle x^2x^*, P_{1,1}(x) \rangle = \frac{1}{\sqrt{\mu_{2,2} - \mu_{1,1}^2}} \mathbb{E} [x^2x^*(xx^* - \mu_{1,1})] = 0$$

d'où :

$$P_{2,1}(x) = x^2x^* - \frac{\mu_{2,2}}{\mu_{1,1}}x$$

Sa norme est :

$$|P_{2,1}(x)|^2 = \mu_{3,3} - \frac{\mu_{2,2}^2}{\mu_{1,1}}$$

Par suite :

$$P_{2,1}(x) = \frac{\mu_{1,1}x^2x^* - \mu_{2,2}x}{\sqrt{\mu_{1,1}^2\mu_{3,3} - \mu_{2,2}^2\mu_{1,1}}}$$

C.2.2 Liens avec les coefficients des noyaux du modèle de Volterra

Selon (4.56),

$$f_{2i+1}(n_1, \dots, n_{2i+1}) = \sum_{j=i}^1 \mathbb{E}[V_{2j+1}^*(U(n)) Q_{n_1, \dots, n_{2i+1}}^{(2i+1)}(U(n))]$$

On a :

$$\begin{aligned} f_3(n_1, n_2, n_3) &= \mathbb{E}[V_3^*(U(n)) Q_{n_1, n_2, n_3}^{(3)}(U(n))] \\ &= \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=0}^{M-1} h_3^*(i, j, k) \mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1, n_2, n_3}^{(3)}(U(n))] \end{aligned} \quad (C.1)$$

Tenant compte de la définition des polynômes $Q_{n_1, n_2, n_3}^{(3)}(U(n))$, on a :

$$\mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1, n_1, n_1}^{(3)}(U(n))] = \begin{cases} \sqrt{\mu_{3,3} - \frac{\mu_{2,2}^2}{\mu_{1,1}}} & \text{si } n_1 = i = j = k \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1, n_2, n_2}^{(3)}(U(n))] = \begin{cases} \sqrt{\mu_{1,1} \mu_{2,2} - \mu_{1,1}^3} & \text{si } \begin{cases} n_1 = i \neq n_2 = j = k \\ n_1 = j \neq n_2 = i = k \end{cases} \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1, n_1, n_3}^{(3)}(U(n))] = \begin{cases} \sqrt{\mu_{2,2} \mu_{1,1}} & \text{si } n_3 = k \neq n_1 = i = j \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1, n_2, n_3}^{(3)}(U(n))] = \begin{cases} \sqrt{\mu_{1,1}^3} & \text{si } \begin{cases} n_1 = i \neq n_2 = j \neq n_3 = k \\ n_1 = j \neq n_2 = i \neq n_3 = k \end{cases} \\ 0 & \text{sinon} \end{cases}$$

En remplaçant ces quantités dans l'équation (C.1), on obtient les relations (4.58).

Pour les termes du premier ordre on a :

$$\begin{aligned} f_1(n_1) &= \mathbb{E}[V_1^*(U(n)) Q_{n_1}^{(1)}(U(n))] + \mathbb{E}[V_3^*(U(n)) Q_{n_1}^{(1)}(U(n))] \\ &= \sum_{i=0}^{M-1} h_1^*(i) \mathbb{E}[u^*(n-i) Q_{n_1}^{(1)}(U(n))] \\ &\quad + \sum_{i=0}^{M-1} \sum_{j=i}^{M-1} \sum_{k=0}^{M-1} h_3^*(i, j, k) \mathbb{E}[u^*(n-i) u^*(n-j) u(n-k) Q_{n_1}^{(1)}(U(n))] \end{aligned} \quad (C.2)$$

Tenant compte de la définition de $Q_{n_1}^{(1)}$, on a :

$$\mathbb{E}[u^*(n-i)Q_{n_1}^{(1)}(U(n))] = \begin{cases} \sqrt{\mu_{1,1}} & \text{si } i = n_1 \\ 0 & \text{sinon} \end{cases}$$

$$\mathbb{E}[u^*(n-i)u^*(n-j)u(n-k)Q_{n_1}^{(1)}(U(n))] = \begin{cases} \frac{\mu_{2,2}}{\sqrt{\mu_{1,1}}} & \text{si } n_1 = i = j = k \\ \sqrt{\mu_{1,1}^3} & \text{si } j = k \neq n_1 = i \\ \sqrt{\mu_{1,1}^3} & \text{si } i = k \neq n_1 = j \\ 0 & \text{sinon} \end{cases}$$

En remplaçant ces quantités dans l'équation (C.2), on obtient les relations (4.59).

Annexe D

Méthodes de Tseng et Powers

Dans cette annexe, nous résumons les méthodes de Tseng et Powers auxquelles nous faisons référence dans le chapitre 4. La première méthode, [131], est consacrée à l'identification d'un modèle de Volterra cubique avec une entrée réelle *i.i.d.*. Nous présentons les différentes équations dans le cas d'une entrée ayant une distribution de probabilité symétrique. La seconde méthode, [129], est consacrée à l'identification d'un modèle de Volterra passe-bande d'ordre trois avec une entrée de type QAM.

D.1 Cas d'un modèle cubique ayant une entrée *i.i.d.* symétrique

$$\begin{aligned} h_3(n_1, n_2, n_3) &= \frac{\mathbb{E}[d(n)u(n-n_1)u(n-n_2)u(n-n_3)]}{\mu_2^3}, \quad n_1 \neq n_2 \neq n_3 \\ h_3(n_1, n_2, n_2) &= \frac{\mu_2 \mathbb{E}[d(n)u(n-n_1)] - \mathbb{E}[d(n)u(n-n_1)u^2(n-n_2)]}{\mu_2^3 - \mu_2 \mu_4}, \quad n_1 \neq n_2 \\ h_3(n_1, n_1, n_2) &= \frac{\mu_2 \mathbb{E}[d(n)u(n-n_2)] - \mathbb{E}[d(n)u(n-n_2)u^2(n-n_1)]}{\mu_2^3 - \mu_2 \mu_4}, \quad n_1 \neq n_2 \\ h_2(n_1, n_2) &= \frac{\mathbb{E}[d(n)u(n-n_1)u(n-n_2)]}{\mu_2^2}, \quad n_1 \neq n_2 \\ H_3(n_1) &= \sum_{n_2=0, n_2 \neq n_1}^{M-1} h_3(n_1, n_2, n_2) \end{aligned}$$

$$\begin{aligned}
 H_3 &= \frac{\mu_2 \sum_{n_1=0}^{M-1} \mathbb{E}[d(n)u^3(n-n_1)] - \mu_4 \sum_{n_1=0}^{M-1} \mathbb{E}[d(n)u(n-n_1)]}{\mu_6\mu_2 - \mu_4^2} \\
 H_2 &= -\frac{\sum_{n_1=0}^{M-1} \mathbb{E}[d(n)u^2(n-n_1)]}{\mu_4 + (M-1)\mu_2^2} \\
 w(n_1) &= \mu_4 \mathbb{E}[d(n)u^2(n-n_1)] - \mu_4\mu_2^2 H_2 \\
 h_3(n_1, n_1, n_1) &= \frac{\mu_2 \mathbb{E}[d(n)u^3(n-n_1)] - \mu_4 \mathbb{E}[d(n)u(n-n_1)]}{\mu_6\mu_2 - \mu_4^2} \\
 h_2(n_1, n_1) &= \frac{w(n_1)}{\mu_4^2 - \mu_4\mu_2^2} \\
 h_1(n_1) &= \frac{\mathbb{E}[d(n)u(n-n_1)] - \mu_2^2 H_3(n_1) - \mu_4 h_3(n_1, n_1, n_1)}{\mu_2}
 \end{aligned}$$

D.2 Cas d'un modèle cubique passe-bande ayant une entrée *i.i.d.* de type QAM

$$\begin{aligned}
 h_3(n_1, n_2, n_3) &= \frac{\mathbb{E}[d(n)u^*(n-n_1)u^*(n-n_2)u(n-n_3)]}{\mu_{1,1}^3} \\
 h_1(n_1) &= \frac{\mathbb{E}[d(n)u^*(n-n_1)] - \mu_{2,2} h_3(n_1, n_1, n_1) - \mu_{1,1}^2 \sum_{n_2=0, n_2 \neq n_1}^{M-1} h_3(n_1, n_2, n_2)}{\mu_{1,1}}
 \end{aligned}$$

Bibliographie

- [1] S. Abeysekera and Z. Zang. Optimal Laguerre filters for sigma-delta demodulator circuits. *Signal Processing*, 80 :205–209, 2000.
- [2] N. Ahmed and D. H. Youn. On a realization and related algorithm for adaptive prediction. *IEEE Trans. on Acoustics, Speech and Signal Processing*, ASSP-28(5) :493–497, October 1980.
- [3] H. Akçay and P.S.C. Heuberger. A frequency domain iterative identification algorithm using general orthonormal basis functions. *Automatica*, 37 :663–664, 2001.
- [4] H. Akçay and B. Ninness. Rational basis functions for robust identification from frequency and time domain measurements. *Automatica*, 34 :1101–1117, 1998.
- [5] A. Alper. A consideration of the discrete Volterra series. *IEEE Trans. Automatic Control*, Vol. AC-10(3) :322–327, July 1965.
- [6] P.O. Amblard, M. Gaeta, and J.L. Lacoume. Statistics for complex variables and signals- Part II : signals. *Signal Processing*, 53(6) :15–25, 1996.
- [7] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear programming : Theory and algorithms*. Wiley, New-York, 2nd edition, 1993.
- [8] S. Benedetto and E. Biglieri. Nonlinear equalization of digital satellite channels. *IEEE J. Select. Areas Communications*, SAC-1 :57–62, January 1983.
- [9] S. Benedetto, E. Biglieri, and E. Daffara. Modeling and performance evaluation of nonlinear satellite links- a Volterra series approach. *IEEE Trans. Aerospace and Electronics Systems*, 15(4) :494–506, July 1979.
- [10] N.J. Bershad, P. Celka, and S. Mc Laughlin. Analysis of stochastic gradient identification of Wiener-Hammerstein systems for nonlinearities with Hermite polynomial expansions. *IEEE Trans. on Signal Processing*, 49(5) :1060–1072, May 2001.
- [11] E. Biglieri, A. Gersho, R. Gitlin, and T. Lim. Adaptive cancellation of nonlinear intersymbol interference for voiceband data transmission. *IEEE J. Select. Areas Commun.*, 2 :765–777, 1984.
- [12] S.A. Billings. Identification of nonlinear systems- a survey. *IEE Proceedings Part D*, 127(6) :272–285, 1980.
- [13] P. Bodin. *On the selection of best orthonormal basis in system identification*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, 1997.

-
- [14] P. Bodin. Recursive calculation of expansion coefficients for generalized orthonormal rational bases. In *Proc. of 14th IFAC World Congress*, pages 373–378, Beijing, China, 1999.
 - [15] P. Bodin, T. Oliveira e Silva, and B. Wahlberg. On the construction of orthonormal basis functions for system identification. In *Proc. of 13th IFAC World Congress, Vol. I*, pages 369–374, San Francisco, 1996.
 - [16] P. Bodin, L. F. Villemoes, and B. Wahlberg. An algorithm for selection of best orthonormal rational basis. In *Proc. of 36th IEEE Conference on Decision and Control (CDC)*, page 1997, 1997.
 - [17] P. Bodin and B. Wahlberg. Fast expansion coefficient calculation for best orthonormal rational basis. In *Proc. MTNS'98*, Padova, Italy, 1998.
 - [18] S. Boyd and L.O. Chua. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Trans. on Circuits and Systems*, 32(11) :1150–1161, 1985.
 - [19] R.J.G.B. Campello. *New architectures and methodologies for modeling and control of complex systems combining classical and modern tools*. PhD thesis, School of Electrical and Computer Engineering of the State University of Campinas (FEEC/UNICAMP), Campinas/SP, Brazil, 2002.
 - [20] R.J.G.B. Campello, W.C. Amaral, and G. Favier. Optimal Laguerre series expansion of discrete Volterra models. In *Proc. of the European Control Conference (ECC)*, Porto, Portugal, September 2001.
 - [21] R.J.G.B. Campello, G. Favier, and W.C. Amaral. Optimal expansions of discrete-time Volterra models using Laguerre functions. *Automatica*, 40 :815–822, 2004.
 - [22] A. Carini, G. Sicuranza, and V.J. Mathews. On the inversion of certain nonlinear systems. *IEEE Signal Processing Letters*, 4 :334–336, 1997.
 - [23] J.D. Carroll and J.J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35 :283–319, 1970.
 - [24] C.-H. Cheng and E.J. Powers. Optimal Volterra kernel estimation algorithms for a nonlinear communication system for PSK and QAM inputs. *IEEE Trans. Signal Processing*, 49(1) :147–163, 2001.
 - [25] H. Chiang, C.L. Nikias, and A.N. Venetsanopoulos. Efficient implementation of quadratic digital filters. *IEEE Trans. Acoust., Speech and Signal Processing*, 34(6) :1511–1528, 1986.
 - [26] J. Choi. Sufficient conditions of the redundancy-transform for the blind identification. *Signal Processing*, 79 :151–160, 1999.
 - [27] P. Comon. Circularité et signaux aléatoires à temps discret. *Traitement du Signal*, 11(5) :417–420, 1994.
 - [28] P. Comon. Independent component analysis, contrasts, and convolutive mixtures. In *Proc. IMA Conf. Mathematics in Communications II*, Lancaster University, UK, 16-18 Dec. 2002.

- [29] C.E. Davila, A.J. Welch, and H.G. Rylander III. A second-order adaptive Volterra filter with rapid convergence. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, ASSP-35(9) :1259–1263, September 1987.
- [30] A. De Baynast and L. De Lathauwer. Détection autodidacte pour des systèmes à accès multiples basée sur l’analyse PARAFAC. In *Proc. of GRETSI’03*, Paris, September 2003.
- [31] T. J. de Hoog, P. S. C. Heuberger, and P. M. J. Van den Hof. A general transform theory of rational basis function expansions. In *Proc. of the 39th Conference on Decision and Control*, 2000.
- [32] Z. Ding. Characteristics of band-limited channels unidentifiable from second-order cyclostationarity statistics. *IEEE Signal Processing letters*, 3 :150–152, May 1996.
- [33] B. Farhang-Boroujny. Pilot- based channel identification : proposal for semi-blind identification of communications channels. *Electronics Letters*, 31(13) :1044–1046, 1995.
- [34] G. Favier. Estimation paramétrique de modèles entrée-sortie. In M. Gugliemi, editor, *Signaux aléatoires : modélisation, estimation, détection*, Traité IC2, chapter 7. Lavoisier, Hermès Science Publications, Paris, 2004.
- [35] G. Favier, A.Y. Kibangou, and R.G.J.B. Campello. Nonlinear system modelling by means of orthonormal basis functions. In *2nd IEEE Conference on Signal, Systems, Decision and Information theory (SSD)*, Sousse, Tunisia, March 2003.
- [36] G. Favier, A.Y. Kibangou, and A. Khouaja. Nonlinear system modelling by means of Volterra models. approaches for the parametric complexity reduction. In *Invited paper in Symposium Techniques Avancées et Stratégies Innovantes en Modélisation et Commande Robuste des processus industriels*, Martigues, France, September 2004. the Instrumentation Systems and Automation Society (ISA).
- [37] A. Fermo, A. Carini, and G.L. Sicuranza. Simplified volterra filters for acoustic echo cancellation in GSM receivers. In *EUSIPCO-2000*, Tampere, Finland, September 2000.
- [38] X.N. Fernando and A. B. Sesay. Higher order adaptive filter based predistorsion for nonlinear distortion compensation of radio over fiber fiber links. In *Proc. of the International Conference on Communications (ICC’ 2000)*, vol. 1/3, pages 367–371, New-Orleans, LA, June 2000.
- [39] X.N. Fernando and A.B. Sesay. Fiber wireless channel estimation using correlation properties of PN sequences. *Canadian Journal of Electrical and Computer Engineering*, 26(2), April 2001.
- [40] X.N. Fernando and A.B. Sesay. A Hammerstein type decision feedback equalizer for the fiber-wireless uplink. In *Proc. of 12th International Conf. on Wireless Communications (Wireless’2001)*, pages 318–325, Calgary, Canada, July 2001.
- [41] Z. Fezjo and H. Lev-Ari. Adaptive nonlinear Wiener-Laguerre- lattice models. In *Proc. of ICASSP*, pages 977–980, Detroit, MI, 1995.

- [42] Z. Fezjo and H. Lev-Ari. Adaptive Laguerre-lattice filters. *IEEE Trans. on Signal Processing*, 45(12) :3006–3016, 1997.
- [43] W. Frank. An efficient approximation to the quadratic Volterra filter and its application in real-time loudspeaker linearization. *Signal Processing*, 45(1) :97–113, 1995.
- [44] Y. Fu and G. A. Dumont. An optimum time scale for discrete Laguerre network. *IEEE Trans. on Automatic Control*, 38 :934–938, 6 1993.
- [45] G. B. Giannakis and E. Serpedin. Linear multichannel blind equalizers of non-linear FIR Volterra channels. *IEEE Trans. Signal Processing*, 45(1) :67–81, January 1997.
- [46] G.B. Giannakis. Filterbanks for blind channel identification and equalization. *IEEE Signal Processing Letters*, 4(1) :184–187, 1997.
- [47] G.B. Giannakis and E. Serpedin. A bibliography on nonlinear system identification. *Signal Processing*, 81(3) :533–580, March 2001.
- [48] G. H. Golub and C. F. Van Loan. *Matrix computations*. The John Hopkins University Press, Baltimore, Maryland, third edition, 1996.
- [49] J.C. Gomez and E. Baeyens. Identification of multivariable Hammerstein systems using rational orthonormal bases. In *Proc. of the 39th IEEE Conf. on Decision and Control (CDC)*, pages 2849–2854, Sydney, Australia, December 2000.
- [50] W. Greblicki. Nonparametric identification of Wiener systems by orthogonal series. *IEEE Trans. Automatic Control*, 39(10) :2077–2086, October 1994.
- [51] M. Green and A. M. Zoubir. Basis sequence approximation for time-varying quadratic Volterra models. In *Proc. of the 8th IEEE Workshop on Intelligent Signal Processing and Communications Systems (ISPACS'00)*, Vol. 2, pages 727–730, Honolulu, Hawaii, November 2000.
- [52] D.W. Griffith and G.R. Arce. Partically decoupled Volterra filters : formulation and LMS adaptation. *IEEE Trans. on Signal Processing*, 45(6) :1485–1494, June 1997.
- [53] S. Gunnarsson and B. Wahlberg. Some asymptotic results in recursive identification using Laguerre models. *Int. Journal of Adaptive Control and Signal Processing*, 5 :313–333, 1991.
- [54] R. Hacioglu and G.A. Williamson. Reduced complexity Volterra models for nonlinear system identification. *EURASIP Journal on Applied Signal Processing*, 4 :257–265, December 2001.
- [55] R Hacioglu and G.A. Williamson. Identification of Wiener systems using reduced complexity Volterra models. In *13th IFAC SYSID*, pages 365–370, Rotterdam, The Netherlands, August 2003.
- [56] R.A. Harshman. Foundation of the PARAFAC procedure : models and conditions for an "explanatory" multimodal factor analysis. *UCLA working papers in phonetics*, 16 :1–84, 1970.

- [57] S. Haykin. *Adaptive filter theory*. Englewood Cliffs, NJ : Prentice-Hall, 2nd edition, 1991.
- [58] P. S. C. Heuberger, P. M. J. Van den Hof, and O. H. Bosgra. A generalized orthonormal basis for linear dynamical systems. *IEEE Trans. on Automatic Control*, 40(3) :451–465, 1995.
- [59] M. Y. Ismail. *Adaptation of generalized feedforward filters with applications to speech*. PhD thesis, University of Florida, 1997.
- [60] D. J. Kang, D. O. Hwang, S. W. Nam, and E. J. Powers. Adaptive Volterra filtering using m-band wavelet transform and Gram-Schmidt orthogonalization. *Electronics Letters*, 38(6) :291–292, March 2002.
- [61] G. Karam and H. Sari. Analysis of predistorsion, equalization and ISI cancellation techniques in digital radio systems with nonlinear transmit amplifiers. *IEEE Trans. on Communication*, 37 :1245–1253, December 1989.
- [62] K. Kazlauskas and J. Kazlauskas. Layered polynomial filter. *Informatica*, 13(1) :23–36, 2002.
- [63] A. Khouaja, G. Favier, and H. Messaoud. System identification using generalized orthonormal bases. In *IEEE Conference on Signal, Systems, Decision and information theory (SSD)*, Sousse, Tunisia, 26-28 March 2003.
- [64] A. Khouaja, A.Y. Kibangou, and G. Favier. Third-order Volterra kernels complexity reduction using PARAFAC. In *IEEE-ISCCSP'04*, Hammamet, Tunisia, March 2004.
- [65] A. Y. Kibangou, G. Favier, and M. M. Hassani. A growing approach for selecting generalized orthogonal basis functions in the context of system modelling. In *Proc. of IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP'03)*, June 2003.
- [66] A.Y. Kibangou, G. Favier, and M.M. Hassani. Selection of generalized orthonormal bases for second order Volterra filters. *Signal Processing (submitted)*.
- [67] A.Y. Kibangou, G. Favier, and M.M. Hassani. Algorithme hiérarchique pour l'estimation optimale des noyaux de volterra développés en séries de laguerre. In *Actes du 6ème Colloque Africain sur la recherche en Informatique, CARI 02*, pages pp. 395–402, Yaoundé, Cameroun, Octobre 2002.
- [68] A.Y. Kibangou, G. Favier, and M.M. Hassani. Generalized orthonormal basis selection for expanding quadratic Volterra filters. In *Proc. IFAC Symposium on System Identification (SYSID)*, pages 1119–1124, Rotterdam, The Netherlands, October 2003.
- [69] A.Y. Kibangou, G. Favier, M.M. Hassani, and A. Zeroual. estimation optimale des noyaux de Volterra développés en séries de Laguerre. In *Conférence internationale de Mathématiques Appliquées aux Sciences de l'Ingénieur (CIMASI)*, Casablanca, Maroc, Octobre 2002.
- [70] T. Koh and E.J. Powers. Second-order Volterra filtering and its application to nonlinear system identification. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, ASSP-33(6) :1445–1455, December 1985.

- [71] M.J. Korenberg. Parallel cascade identification and kernel estimation for nonlinear systems. *Annals of Biomedical Engineering*, 19 :429–455, 1991.
- [72] P. Koukoulas and N. Kalouptsidis. Blind identification of second order Hammerstein series. *Signal Processing*, 83 :213–234, 2003.
- [73] J.B. Kruskal. Three-way arrays : rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra Applicat* ;, 18 :95–138, 1977.
- [74] F. Kuech and W. Kellermann. Coefficient-dependent step-size for adaptive second-order Volterra filters. In *Proc. of EUSIPCO*, pages 1805–1808, Wien, Austria, September 2004.
- [75] J.L. Lacoume. Variables et signaux aléatoires complexes. *Traitement du Signal*, 15(6) :535–544, 1998.
- [76] J.M. Le Caillec and R. Garello. Time series nonlinearity modeling : A Giannakis formula type approach. *Signal Processing*, 83 :1759–1788, 2003.
- [77] J. Lee and V.J. Mathews. A fast recursive least squares adaptive second-order Volterra filter and its performance analysis. *IEEE Trans. Signal Processing*, 41(3) :1087–1102, March 1993.
- [78] R. Lopez-Valcarce and S. Dasgupta. Blind equalization of nonlinear channels from second-order statistics. *IEEE Trans. Signal Processing*, 49(12) :3084–3097, 2001.
- [79] R. Lopez-Valcarce, X. Song, and S. Dasgupta. Blind equalization of nonlinear channels from second-order statistics using precoding and channel diversity. In *Proc. ICASSP*, vol. 4, pages 2073–2076, Salt-Lake city, UT, USA, 2001.
- [80] R.W. Lucky. Modulation and detection for data transmission on the telephone channel. In J.K. Skwirzynski, editor, *New directions in signal processing in communication and control*. Leiden, Holland, 1975.
- [81] L. Ludeman. Blind identification of instantaneous nonlinear polynomial systems. In *Proc. IEEE-EURASIP Workshop Nonlinear Signal and Image Processing, NSIP'95*, Halkidiki, Greece, June 20-22 1995.
- [82] R. Malti. *Représentation de systèmes discrets sur la base des filtres orthogonaux : Application à la modélisation des systèmes multivariables*. PhD thesis, INPL, Nancy, France, 1999.
- [83] D.P. Mandic, A.I. Hanna, and M. Razaz. A normalized gradient descent algorithm for nonlinear adaptive filters using a gradient adaptive step size. *IEEE Signal Processing letters*, 8(11) :295–297, November 2001.
- [84] M. A. Masnadi-Shirazi and N. Ahmed. Optimum Laguerre networks for a class of discrete-time systems. *IEEE Trans. on Signal Processing*, 39(9), 1991.
- [85] M. A. Masnadi-Shirazi and M. Ghasemi. Adaptive Laguerre network realization. *Signal Processing*, 80 :2169–2186, 2000.
- [86] V.J. Mathews. Adaptive polynomial filters. *IEEE Signal Processing Mag.*, 8(3) :10–26, 1991.

- [87] R. Merched and A. Sayed. Order-recursive RLS Laguerre adaptive filtering. *IEEE Trans. on Signal Processing*, 48(10) :3000–3010, 2000.
- [88] R. Merched and A. Sayed. RLS Laguerre lattice adaptive filtering :error feed-back and array-based algorithms. In *Proc. of ISCAS, Vol. 2*, pages 669–672, Sydney, Australia, 2001.
- [89] L. Ngia. *System Modeling Using Basis Functions and Application to Echo Cancellation*. PhD thesis, Department of Signals and Systems, School of Electrical and Computer Engineering , Chalmers university of technology, Göteborg, Sweden, October 2000.
- [90] B. Ninness and F. Gustafsson. A unifying construction of orthonormal bases for system identification. *IEEE Trans. on Automatic Control*, pages 515–521, 1997.
- [91] B. Ninness and H. Hjalmarsson. The fundamental role of general orthonormal bases in system identification. *IEEE Trans. Automatic Control*, 44(7) :1384–1406, 1999.
- [92] R. Nowak and B.D. Van Veen. Volterra filter equalization : A fixed point approach. *IEEE Trans. Signal Processing*, 45 :377–388, 1997.
- [93] R.D. Nowak and B.D. Van Veen. Random and pseudorandom inputs for Volterra filter. *IEEE Trans. Signal Processing*, 42(8) :2124–2135, 1994.
- [94] T. Oliveira e Silva. On the determination of the optimal pole position of Laguerre filters. *IEEE Trans. Signal Processing*, 43(9), 1995.
- [95] T. Oliveira e Silva. Optimality conditions for Laguerre lattice filters. *IEEE Signal Processing letters*, 2(5) :342–346, 1995.
- [96] T. Oliveira e Silva. Optimality conditions for truncated Kautz networks with two periodically repeating complex conjugate poles. *IEEE Trans. on Automatic Control*, 40(2), 1995.
- [97] T. Oliveira e Silva. Stationarity conditions for the l2 error surface of the generalized basis functions lattice filter. *Signal Processing*, 56 :233–253, 1997.
- [98] T. Oliveira e Silva. Optimal pole conditions for Laguerre and two parameters Kautz models : A survey of known results. In *Proc. of IFAC- SYSID, Vol. 2*, pages 457–462, Santa-Barbara, California, 2000. Pergamon.
- [99] A.V. Oppenheim and R.W. Schaffer. *Discrete-time signal processing*. Prentice-Hall, Inc., Englewood Cliffs, New-Jersey 07632, 1989.
- [100] M. J. L. Orr. Introduction to radial basis function networks. Technical report, Centre for cognitive science, University of Edinburgh, Scotland, 1996.
- [101] M. T. Ozden, A.H. Kayran, and E. Panayirci. Adaptive Volterra filtering with complete lattice orthogonalization. *IEEE Trans. on Signal Processing*, 44(8) :2092–2098, August 1996.
- [102] G. Palm. On representation and approximation of nonlinear systems. *Biological Cybernetics*, 34 :49–52, 1979.
- [103] G. Panda, S.K. Maharatha, and R.R. Mishra. Adaptive blind identification of the parameters of nonlinear systems using higher order statistics. In *Proc. of ICSPAT'96*, pages 200–204, Boston, USA, Oct. 29-31 1996.

- [104] T.M. Panicker and V.J. Mathews. Parallel-cascade realizations and approximations of truncated volterra systems. *IEEE Trans. on Signal Processing*, 46(10) :2829–2831, October 1998.
- [105] T.M. Panicker, V.J. Mathews, and G.L. Sicuranza. Adaptive parallel-cascade truncated volterra filters. *IEEE Trans. on Signal Processing*, 46(10) :2624–2673, October 1998.
- [106] V. N. Parikh and A. Z. Baraniecki. The use of the modified escalator algorithm to improve the performance of transform-domain LMS adaptive filters. *IEEE Trans. on Signal Processing*, 46(3) :625–635, 1998.
- [107] H. Perez and S. Tsujii. A system identification using orthogonal functions. *IEEE Trans. on Signal Processing*, 39(3), 1991.
- [108] N. Petrochilos and P. Comon. Nonlinear channel identification and performance analysis. In *Proc. ICASSP, vol.1*, pages 209–212, Istanbul, Turkey, June 2000.
- [109] G. Picard and O. Cappé. Blind identification of Hammerstein nonlinear distortion models. In *Proc. of the IEEE workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, USA, October, 19-22 2003.
- [110] S. Prakriya and D. Hatzinakos. Blind identification of linear subsystems of LTI-ZMNL-LTI models with cyclostationary inputs. *IEEE Trans. on Signal Processing*, 45(8) :2023–2036, August 1997.
- [111] S. Pupolin and L. Greenstein. Performance analysis of digital radio links with nonlinear transmit amplifiers. *IEEE Journal on Sel. Area in Communications*, 37 :534–546, April 1987.
- [112] G. V. Raz and B. Van Veen. Baseband Volterra filters for implementing carrier based nonlinearities. *IEEE Trans. Signal Processing*, 46(1) :103–114, January 1998.
- [113] G. V. Raz and B. Van Veen. Blind equalization and identification of nonlinear and IIR systems- A least squares approach. *IEEE trans. Signal Processing*, 48(1) :192–200, January 2000.
- [114] A.J. Redfern and G. Tong Zhou. Nonlinear channel identification and equalization for OFDM systems. In *Proc. IEEE ICASSP’98, vol. 6*, pages 3521–3524, May 1998.
- [115] A.J. Redfern and G. Tong Zhou. A root method for Volterra system equalization. *IEEE Signal Processing Letters*, 5(5) :285–288, 1998.
- [116] A.J. Redfern and G. Tong Zhou. Multichannel and block based precoding methods for fixed point equalization of nonlinear communication channels. *Signal Processing*, 81 :1041–1052, 2001.
- [117] R. A. Roberts and C. T. Mullis. *Digital Signal Processing*. Addison-Wesley series in Electrical Engineering : Digital Signal Processing. Addison-Wesley Publishing Company, Reading, MA, 1987.
- [118] A. M. Saleh. Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers. *IEEE Trans. Communications*, 29(11) :1715–1720, November 1981.

- [119] A. Sarti and S. Pupolin. Recursive techniques for the synthesis of a p th-order inverse of a Volterra system. *Europ. Trans. Telecommun. Related Technol.*, 3 :315–322, 1992.
- [120] M. Schetzen. Theory of p th-order inverses of nonlinear systems. *IEEE Trans. on Circuits and Systems*, CAS-23(5) :285–291, May 1976.
- [121] M. Schetzen. *The Volterra and Wiener theories of nonlinear systems*. John Wiley and Sons, Inc., New-York, 1980.
- [122] N.D. Sidiropoulos, G.B. Giannakis, and R. Bro. Blind PARAFAC receivers for DS-CDMA systems. *IEEE Trans. Signal Processing*, 48(3) :810–823, March 2000.
- [123] M.A. Syed and V.J. Mathews. Lattice algorithms for adaptive Volterra filtering. *IEEE Trans. Circuits and Systems II : Analog and Digital Signal Processing*, 41(3) :202–214, March 1994.
- [124] L. Tan and J. Jiang. Adaptive Volterra filters for active control of nonlinear noise processes. *IEEE Trans. Signal Processing*, 49(8) :1667–1676, August 2001.
- [125] N. Tanguy, R. Morvan, P. Vilbé, and L. C. Calvez. Online optimization of the time scale in adaptive Laguerre-based filters. *IEEE Trans. on Signal Processing*, 48(4), 2000.
- [126] N. Tanguy, P. Vilbé, and L. C. Calvez. Optimum choice of free parameter in orthonormal approximations. *IEEE Trans. on Automatic Control*, 40(10), 1995.
- [127] G. Tong L., Xu and T. Kailath. Blind identification and equalization based on second-order statistics : A time-domain approach. *IEEE Trans. Inform. Theory*, 40(3) :340–350, 1994.
- [128] G. Tong Zhou and G.B. Giannakis. Nonlinear channel identification and performance analysis for PSK inputs. In *Proc. 1st IEEE Signal Processing Workshop on Wireless Communications*, pages 337–340, Paris, France, April 1997.
- [129] C.-H. Tseng and E.J. Powers. Identification of nonlinear channels in digital transmission systems. In *Proc. of IEEE Signal Processing Workshop Higher-order Statistics*, pages pp. 42–45, South Lake Tahoe, CA, June 1993.
- [130] C.-H. Tseng and E.J. Powers. Nonlinear channel equalization in digital satellite systems. In *Proc. IEEE Global telecommunications Conf. (GLOBECOM)*, pages 1639–1643, Houston, TX, November 1993.
- [131] C.-H. Tseng and E.J. Powers. Identification of cubic systems using higher order moments of I.I.D. signals. *IEEE Trans. Signal Processing*, 43(7) :1733–1735, 1995.
- [132] P. M. J. Van den Hof, P. S. C. Heuberger, and J. Bokor. System identification with generalized orthonormal basis functions. *Automatica*, 31(12) :1821–1834, 1995.
- [133] B. Wahlberg. Laguerre and Kautz models. In *Invited Survey at IFAC-SYSID'94*, Vol. 3, pages 1–12, Copenhagen, July 1994.

- [134] B. Wahlberg. Orthonormal basis functions models : A transformation analysis. In *Proc. of 14th IFAC World Congress, Vol. H*, pages 355–359, Beijing, China, 1999.
- [135] B. Wahlberg and P. Mäkilä. On approximation of stable linear systems using Laguerre and Kautz functions. *Automatica*, 32(5) :693–708, 1996.
- [136] W.I. Way. Optical fiber based microcellular systems. an overview. *IEICE Trans. Commun.*, E76-B(9) :1091–1102, September 1993.
- [137] D.T. Westwick. *Methods for the identification of multiple-input nonlinear systems*. PhD thesis, Department of Electrical Engineering and Biomedical Engineering department, Mc Gill University, Montreal, Canada, June 1995.
- [138] G. A. Williamson and S. Zimmermann. Globally convergent adaptive IIR filters based on fixed pole locations. *IEEE Trans. on Signal Processing*, 44(6) :1418–1427, 1996.
- [139] G. Xu, H. Liu, L. Tong, and T. Kailath. A least-squares approach to blind channel identification. *IEEE Trans. Signal Processing*, 43(12) :2982–2993, 1995.