



MASTER 2 SCIENCES DE L'INGÉNIEUR
Université Pierre et Marie Curie - Paris 6

PARCOURS ATIAM

RAPPORT DE STAGE DE FIN D'ÉTUDE

**Séparation en source et système en séries de Volterra :
application à la synthèse sonore avec variation de timbre**

Auteur

Marc WIJNAND

Lieu du stage

Ircam - Équipe S3AM
(s3am.ircam.fr)

Responsable

Thomas HÉLIE - Ircam
(thomas.helie@ircam.fr)

Encadrants

Damien BOUVIER - Ircam
(damien.bouvier@ircam.fr)

David ROZE - Ircam
(david.roze@ircam.fr)

20 août 2017

Résumé

Ce stage se consacre à un problème de l'identification simultanée d'un système et de sa source à partir de la seule donnée de leur sortie. Ce problème est motivé par la synthèse sonore construite à partir d'un modèle identifié sur une base de données.

En première approximation, certains instruments de musique peuvent être décrits par un modèle linéaire autorégressif, correspondant à un filtre résonant tout-pôles, excité par une source. L'algorithme LPC (Codage Prédicatif Linéaire, de l'anglais *Linear Predictive Coding*) existant permet par exemple d'identifier un tel système. Dans ce stage, on s'intéresse au cas où le système peut représenter une évolution du timbre en fonction de l'amplitude, due à des effets non linéaires présents dans les instruments. Des méthodes d'identification de systèmes non linéaires doivent donc être établies.

D'abord, l'algorithme LPC que l'on peut utiliser pour la séparation en source et système linéaire tout-pôle est étendue à un cas non linéaire, un modèle quadratique en l'état, pour faire une séparation en source et système non linéaire à mémoire. Nous montrons que la précision de cette méthode n'est pas suffisante pour l'application envisagée où l'on dispose d'un nombre réduit de données. Deuxièmement, une méthode plus précise qui incorpore une régularisation est établie. Cependant, nous n'avons pu valider cette méthode régularisée.

Mots-clés : identification, systèmes non linéaires, LPC, régularisation, synthèse sonore

Abstract

The subject of this internship is the simultaneous estimation of a system and its source based on the sole knowledge of their output. This problem is encountered when one wants to use a model that has been identified using a sound database for sound synthesis.

A first approach is to approximately represent certain types of musical instruments by a linear, autoregressive model, that corresponds to a resonant all-pole filter, excited by source. Such a system can be identified using the existent LPC (for *Linear Predictive Coding*) algorithm. Since one can observe a timbre evolution for a changing amplitude due to nonlinear effects present in real instruments, there is a need to establish nonlinear system identification methods.

First, the LPC algorithm that can be used for a separation in a source and a linear all-pole system is extended to a nonlinear case, i.e. a model that has a quadratic term in the state. This algorithm will thus identify a source and a nonlinear model with memory. We show that the precision of this method is insufficient for the proposed application where we dispose of a limited number of data. Secondly, regularisation is added in order to establish a more precise identification method. Nevertheless, we have not been able to validate this regularized identification method.

Keywords : identification, nonlinear systems, LPC, regularization, sound synthesis

Remerciements

Je tiens à remercier mes encadrants, Thomas Hélie, Damien Bouvier et David Roze, qui m'ont guidé pendant ce stage, de leur engagement et enthousiasme.

Je remercie également Michel Kieffer pour les explications sur l'analyse par intervalle, et Jean-François Giovannelli, Marco C. Campi et Robert M. Gray de m'avoir envoyé des documents.

Merci aux collègues et professeurs du master ATIAM, à l'équipe S3AM et au personnel et aux chercheurs de l'Ircam de la bonne ambiance de travail.

Finalement, je remercie mes parents et mes sœurs du soutien et de la relecture diligente de ce travail.

Table des matières

Résumé

Remerciements

Notations

Travaux existants

1	Introduction	1
1.1	Organisme d'accueil	1
1.1.1	Ircam	1
1.1.2	UMR STMS	1
1.1.3	Équipe S3AM	2
1.2	Sujet	2
1.2.1	Contexte	2
1.2.2	Problème posé	2
1.2.3	Organisation du rapport de stage	3
1.3	État de l'art	3
1.3.1	Identification de systèmes	4
1.3.2	Codage prédictif linéaire (<i>LPC</i>)	4
1.3.3	Régularisation de problèmes linéaires	5
1.3.4	Synthèse sonore	5
2	Outils de base	7
2.1	Présentation de la méthode LPC	7
2.1.1	Cadre et hypothèses requises	7
2.1.2	Estimation des paramètres	8
2.2	Modèle non linéaire en représentation d'état	9
2.2.1	Modèle non linéaire général	9
2.2.2	Modèles considérés dans ce stage	10
2.3	Séries de Volterra	11
2.3.1	Introduction	11
2.3.2	Lien avec la représentation d'état	11
2.3.3	Noyaux de transfert	11

Contributions

3	Extension de la méthode LPC à un modèle quadratique (QPC)	13
3.1	Présentation de la méthode QPC	13
3.1.1	Cadre et hypothèses requises	13
3.1.2	Estimation des paramètres	13
3.2	Évaluation de la méthode	15
3.2.1	Protocole	16
3.2.2	Résultats	17
3.3	Discussion	18
4	Régularisation de l'estimation	21
4.1	Introduction	21
4.1.1	Principe général de la régularisation	21
4.1.2	Méthode existante pour le cas d'un système linéaire : lisseur du spectre	22
4.1.3	Extension au modèle quadratique : lisseur des noyaux de transfert de Volterra	22
4.1.4	Calcul des hyperparamètres	23
4.2	Méthode avec régularisation du noyau d'ordre 1 (R ₁ QPC)	23
4.2.1	Présentation de la méthode	23
4.2.2	Évaluation	26
4.3	Méthode avec régularisation des noyaux d'ordres 1 et 2 (R ₁₂ QPC)	28
4.3.1	Présentation de la méthode	28
4.3.2	Évaluation	30
5	Test sur des corpus sonores cohérents	31
5.1	Identification d'un corpus sonore cohérent	31
5.2	Synthèse sonore	32
6	Conclusion et perspectives	33
6.1	Conclusion	33
6.2	Perspectives	33
A	Développement de la Méthode LPC	35
A.1	Expression de la vraisemblance	35
A.2	Maximisation de la vraisemblance	36
B	Développement de la Méthode QPC	39
B.1	Expression de la vraisemblance	39
B.2	Maximisation de la vraisemblance	40
C	Idées pour l'amélioration de la méthode QPC	43
C.1	Analyse par intervalle	43
C.2	Information de Fisher et borne de Cramer-Rao	43
C.2.1	Introduction	43
C.2.2	Calcul pour le modèle \mathcal{M}_1	44
C.2.3	Calcul pour le modèle \mathcal{M}_2	45
D	Région de convergence	47

E Développement de la Méthode RQPC	49
E.1 Développement de la Méthode R_1 QPC	49
E.1.1 Calcul du noyau de transfert de Volterra d'ordre 1	49
E.1.2 A priori de lisseur	50
E.1.3 Expression de la vraisemblance	51
E.1.4 Maximisation de la vraisemblance	52
E.1.5 Estimation de l'hyperparamètre λ_1	54
E.2 Développement de la Méthode R_{12} QPC	55
E.2.1 Calcul du noyau de transfert de Volterra d'ordre 2	55
E.2.2 A priori de lisseur	56
E.2.3 Expression de la vraisemblance	57
E.2.4 Maximisation de la vraisemblance	58
E.2.5 Estimation des hyperparamètres λ_1 et λ_2	59
Bibliographie	60
Liste des figures	64
Liste des tableaux	65

Notations

Mathématiques

Algèbre

z^*	Conjugué complexe du nombre $z \in \mathbb{C}$
\mathbf{a}	Vecteur
\mathbf{a}_i	Élément d'un vecteur
\mathbf{o}_n	Vecteur nul de taille $n \in \mathbb{N}^*$
$\ \mathbf{x}\ _p$	Norme p du vecteur \mathbf{x}
\mathbf{A}	Matrice
$\mathbf{A}_{i,j}$	Élément d'une matrice
\mathbf{A}^\top	Transposée de la matrice \mathbf{A}
\mathbf{A}^H	Matrice conjuguée hermitienne de la matrice $\mathbf{A} \in \mathbb{C}^{p \times q}$
\mathbf{I}_n	Matrice unité de taille $n \times n$ avec $n \in \mathbb{N}^*$
$\text{Tr}(\mathbf{A})$	Trace de la matrice carrée \mathbf{A}
\mathbf{O}	Matrice nulle

Analyse

$x \mapsto y = f'(x)$	Fonction dérivée de la fonction $x \mapsto y = f(x)$ par rapport à l'argument x
$y \mapsto x = f^{-1}(y)$	Fonction inverse de la fonction $x \mapsto y = f(x)$
$x \mapsto y = (g \circ f)(x)$	Composition des fonctions $g()$ et $f() : g(f(x))$
$u[n]$	Signal discret u évalué à temps $n \in \mathbb{Z}$

Probabilités

$s \sim \mathcal{N}(\mu, \sigma^2)$	Variable aléatoire qui suit une distribution gaussienne de moyenne μ et d'écart type σ
$p(A)$	Probabilité d'observer un évènement A
$p(A B)$	Probabilité conditionnelle d'observer un évènement A étant donné que l'évènement B s'est produit
$E(s)$	Espérance du variable s
$\text{Cov}(\mathbf{s})$	Matrice de covariance du variable \mathbf{s}

Théorie des ensembles

\mathbb{N}	Ensemble des nombres naturels
\mathbb{N}^*	Ensemble des nombres naturels non nuls
\mathbb{Z}	Ensemble des nombres entiers
$\llbracket a, b \rrbracket_{\mathbb{N}}$	Sous-ensemble de \mathbb{N} , allant de $a \in \mathbb{N}$ à $b \in \mathbb{N}$ (inclus) : $\{a, a+1, \dots, b-1, b\}$
\mathbb{R}	Ensemble des nombres réels
\mathbb{R}^*	Ensemble des nombres réels non nuls

\mathbb{R}_+^*	Ensemble des nombres réels strictement positifs
\mathbb{R}^n	Ensemble des vecteurs à éléments réels de taille $n \in \mathbb{N}^*$
$\mathbb{R}^{p \times q}$	Ensemble des matrices à éléments réels de taille $p \times q$, avec $(p, q) \in \mathbb{N}^* \times \mathbb{N}^*$
\mathbb{C}	Ensemble des nombres complexes
$\mathbb{C}^{p \times q}$	Ensemble des matrices à éléments complexes de taille $p \times q$, avec $(p, q) \in \mathbb{N}^* \times \mathbb{N}^*$
Autres	
\triangleq	Égal par définition

Variables

u	Signal d'entrée
y	Signal de sortie
$\mathbf{a} \in \mathbb{R}^{2K}$	Vecteur des coefficients auto-régressifs
$\mathbf{l} \in \mathbb{R}^{2K^2+K}$	Vecteur des coefficients quadratiques
$\mathbf{L} \in \mathbb{R}^{2K \times 2K}$	Matrice des coefficients quadratiques
$k \in \mathbb{N}$	Ordre de régularisation
$K \in \mathbb{N}^*$	Nombre de résonances
$N \in \mathbb{N}^*$	Longueur de la trame y
$\lambda_i \in \mathbb{R}$	Hyperparamètre de régularisation
$\sigma \in \mathbb{R}_+^*$	Écart type de l'entrée
\mathbb{T}	Intervalle d'indices temporels, $\mathbb{T} \subset \mathbb{N}$

Acronymes

LPC	Codage Prédicatif Linéaire (de l'anglais <i>Linear Predictive Coding</i>)
QPC	Codage Prédicatif Quadratique (de l'anglais <i>Quadratic Predictive Coding</i>)
RQPC	Codage Prédicatif Quadratique Régularisé (de l'anglais <i>Regularized Quadratic Predictive Coding</i>)
GCV	Validation Croisée Généralisée (de l'anglais <i>Generalized Cross Validation</i>)

Chapitre 1

Introduction

Après une présentation de l'organisme d'accueil (Section 1.1), le sujet du stage est introduit (Section 1.2), suivi par l'état de l'art (Section 1.3).

1.1 Organisme d'accueil

Ce stage a été effectué au sein de l'équipe *Systèmes et Signaux Sonores : Audio/Acoustique, instruMents (S3AM)* à l'Ircam.

1.1.1 Ircam

Les origines de l'*Ircam (Institut de Recherche et Coordination Acoustique/Musique)*¹ datent de 1970, mais c'est seulement en 1977 que l'institut est né dans un bâtiment de l'architecte Renzo Piano, avec le compositeur Pierre Boulez comme directeur. Cet institut a la particularité de combiner la recherche scientifique et la création musicale. La recherche scientifique, détaillée dans la Sous-section 1.1.2, peut aboutir à des produits commerciaux. La création musicale consiste en l'assistance de compositeurs pour l'utilisation de nouvelles technologies et à l'organisation de concerts. D'autres activités de transmission sont la documentation, la diffusion de logiciels et l'organisation d'un cursus international de composition et d'un master en acoustique, traitement du signal et informatique, appliqués à la musique.

1.1.2 UMR STMS

L'Ircam accueille l'*Unité Mixte de Recherche Sciences et Technologies de la Musique et du Son (UMR STMS)*, qui a pour tutelles le CNRS (Centre National de la Recherche Scientifique), l'UPMC (Université Pierre et Marie Curie), l'Ircam et le ministère de la Culture et de la Communication. L'UMR réunit une centaine de chercheurs qui mènent une recherche multidisciplinaire sur le son et la musique. Des thèmes de cette recherche incluent le traitement de signal pour l'analyse et la synthèse du son, l'étude de systèmes musicaux, l'acoustique instrumentale, l'analyse et la synthèse de champs acoustiques, la perception et le design sonore, l'interaction son-musique-mouvement, la composition, l'improvisation et la performance assistées par ordinateur.

1. <http://www.ircam.fr>

1.1.3 Équipe S3AM

L'équipe de recherche *S3AM* (pour *Systèmes et Signaux Sonores : Audio/Acoustique, instruMents*)², dirigée par Thomas Hélie, s'intéresse aux systèmes multi-physiques et aux signaux sonores que ces systèmes produisent. On y étudie la modélisation, la simulation, l'identification, l'optimisation, le contrôle, et la conception de systèmes. Une attention particulière est portée aux instruments de musique, à la voix et aux systèmes électroniques audio. Les applications visées concernent les domaines scientifiques, artistiques, pédagogiques et de la santé.

1.2 Sujet

Ce stage porte sur la séparation en source et système en séries de Volterra, avec une application à la synthèse sonore. Premièrement, source et système à mémoire sont identifiés séparément selon un modèle prédéfini ; deuxièmement, un algorithme de synthèse sonore est créé à partir de ces modèles estimés.

1.2.1 Contexte

De nombreux sons résultent de la modification d'une excitation par un système dynamique à mémoire, comme représenté schématiquement en Figure 1.1. On peut citer par exemple des instruments de musique de type impulsif, comme le piano, les cordes pincées et le Fender-Rhodes, ou auto-entretenus, comme les cuivres et les cordes frottées.

Ce découpage en source et système permet une synthèse sonore séparée : celle de la source d'une part, et la simulation du système considéré. Pour cela, il est nécessaire d'obtenir des modèles de la source et du système représentant l'instrument considéré.

Dans ce stage, nous proposons d'identifier ces modèles à partir d'un ou plusieurs signaux audio.

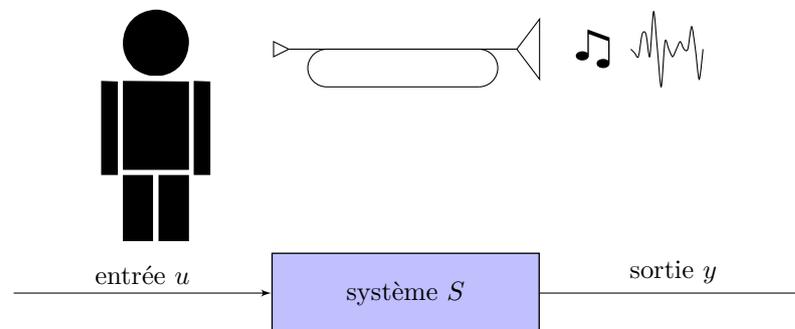


FIGURE 1.1 – Modélisation en source et système dynamique d'un instrument de musique

1.2.2 Problème posé

Il existe plusieurs méthodes d'identification, ayant chacune ses particularités. Dans le cadre de ce stage, les propriétés suivantes sont recherchées.

2. <http://s3am.ircam.fr>

- (a) **Données à disposition** La méthode ne nécessite pas la donnée de la source.
- (b) **Réalisme du timbre** La méthode prend en compte les changements de timbre de l'instrument.
- (c) **Synthèse sonore** La méthode est basée sur un modèle permettant une synthèse rapide (voire temps-réel).

Nous pouvons donc apporter les éléments suivants à notre démarche, qui seront détaillés dans l'état de l'art.

- (a) **Données à disposition** Pour identifier un système à partir de sa sortie seulement, donc sans une connaissance de l'entrée correspondante, on peut se baser sur la méthode LPC. Cette méthode permet de faire la séparation en source et système linéaire à mémoire.
- (b) **Réalisme du timbre** Des changements de timbre d'un instrument sont dus à des effets non linéaires. Par conséquent, il nous faut un modèle non linéaire pour pouvoir les capturer, et une méthode d'identification non linéaire associée.
- (c) **Synthèse sonore** La représentation d'état avec des équations de récurrence et les séries de Volterra permettent de faire une synthèse rapide.

1.2.3 Organisation du rapport de stage

Après un état de l'art à la fin du Chapitre 1, le Chapitre 2 introduit les outils de base que sont la LPC et les séries de Volterra. Dans le Chapitre 3, une extension de la méthode LPC à l'identification d'un modèle quadratique est présentée et évaluée. Le Chapitre 4 ajoute une régularisation à la méthode proposée, dans le but d'améliorer sa robustesse. Finalement, cette dernière méthode est utilisée pour l'identification de corpus sonores dans le Chapitre 5, et la conclusion du stage est donnée dans le Chapitre 6.

1.3 État de l'art

À faibles niveaux d'excitation, l'hypothèse de linéarité du système peut être faite, et le système à mémoire peut être représenté par un filtre convolutionnel, qui est totalement déterminé par (par exemple) une fonction de transfert [1]. Il existe alors des méthodes d'identification associées : des méthodes d'identification entrée-sortie [2] et d'autres comme la LPC (de l'anglais *Linear Predictive Coding* pour Codage Prédicatif Linéaire), développée initialement pour la voix [3].

Dans ce stage, nous proposons d'étendre l'utilisation de la LPC à des systèmes décrits en « source – système non linéaire à mémoire ». Avec cette modélisation, on ne cherche pas à déterminer un ensemble de filtres indexés par l'intensité de la source, comme dans [4], mais une extension plus générale non linéaire qui capture aussi les distorsions et intermodulations.

Le modèle de système utilisé se base sur la représentation d'état [1, 5]. Sous certaines conditions (de non-linéarités régulières et de mémoire à horizon fini), il peut aussi être représenté en séries de Volterra [6, 7] ; on pourra alors bénéficier des avancées de ce formalisme pour la synthèse temps-réel [8, 9, 10].

Dans cet état de l'art, les principes de l'identification sont tout d'abord mentionnés (Sous-section 1.3.1), suivi par une introduction à une méthode d'identification particulière sur laquelle

se base le travail développé dans ce stage, la LPC (Sous-section 1.3.2). Dans la Sous-section 1.3.3, le principe de régularisation est présenté. Finalement, le contexte de la synthèse sonore est illustré en Sous-section 1.3.4.

1.3.1 Identification de systèmes

L'identification d'un système consiste à construire un modèle mathématique d'un système inconnu basé sur la connaissance de certaines données de signaux; elle diffère en cela de la modélisation physique. Il existe beaucoup de méthodes d'identification, qui diffèrent en fonction du type de données nécessaires, des hypothèses sur la structure du système et du fait qu'elles soient paramétriques ou non [2].

Les méthodes d'identification peuvent être séparées en deux catégories (cf. Figure 1.2). Les méthodes où l'entrée u et la sortie y sont connues (Figure 1.2a) appartiennent à une première catégorie. La méthode en déduit un modèle \tilde{S} pour le système S inconnu. Dans cette catégorie, les méthodes NARMAX [11] sont très utilisées. Un autre exemple sont les méthodes qui utilisent un balayage en fréquence comme entrée pour une identification sous l'hypothèse d'un modèle de type Hammerstein [12, 13, 14].

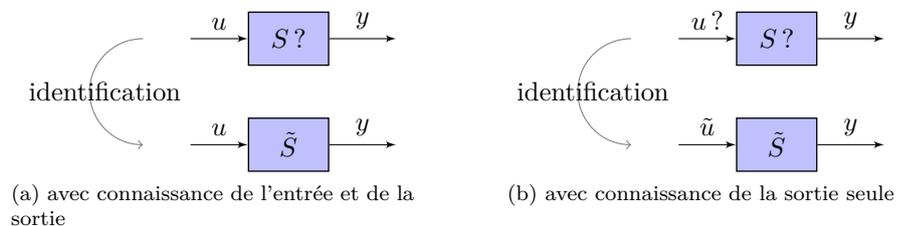


FIGURE 1.2 – Deux catégories d'identification

Si l'on veut identifier une base de données de sons, seulement la sortie y est connue. Ce problème correspond à des méthodes d'une deuxième catégorie (Figure 1.2b). Dans ces méthodes, des hypothèses a priori sur l'entrée u sont faites. Les méthodes développées estiment alors à la fois le système S inconnu et l'entrée u inconnue. La méthode LPC, qui sera présentée en Sous-section 1.3.2, fait partie de cette deuxième catégorie de méthodes d'identification.

1.3.2 Codage prédictif linéaire (LPC)

Issue de la télécommunication, la méthode du codage prédictif linéaire (LPC pour *Linear Predictive Coding*, [15, 3, 16, 17]) a été développée pour l'analyse des signaux de parole. L'idée initiale est d'encoder une trame de signal avec un minimum de paramètres, ce qui permet de réduire la quantité d'information nécessaire pour transmettre la voix [3]. Les origines de cette méthode remontent à 1966 et la LPC est devenue une méthode d'analyse puissante utilisée par exemple dans la voix sur IP développée en 1982 [3].

Pendant la durée d'une trame, la voix est considérée comme le filtrage d'une source suivant une distribution gaussienne de moyenne nulle et d'écart type σ par un filtre linéaire auto-régressif. Ceci est représenté dans la Figure 1.3.

La méthode LPC permet donc de séparer un signal de sortie en une source stochastique et un filtre linéaire tout-pole qui le modifie [16]. Dans ce contexte, des formules pour l'estimation des paramètres de la séparation en source stochastique et filtre linéaire dans le cas de l'identification de sons instrumentaux seront établies dans le Chapitre 2.

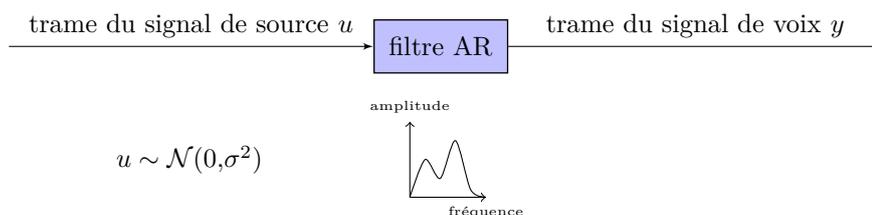


FIGURE 1.3 – Principe de la LPC

1.3.3 Régularisation de problèmes linéaires

Notre but est d'identifier des sons d'une base de données qui consistent en un nombre fini d'échantillons, correspondant typiquement à quelques secondes. Si la méthode développée utilise des estimateurs asymptotiques, il est nécessaire d'avoir un grand nombre de données pour avoir une estimation précise.

Ne disposant que d'un nombre limité de données, il faut utiliser le maximum de l'information présente dans ces données [18]. Une manière d'améliorer une estimation de paramètres d'un modèle linéaire est d'inclure une régularisation dans la fonction de coût, par exemple une distribution a priori sur les paramètres. Dans [19, 20] par exemple, une distribution a priori est choisie correspondante à un critère de spectre lisse.

1.3.4 Synthèse sonore

La synthèse sonore est la génération de sons par des appareils numériques à l'aide de la conversion numérique-analogique.

Il existe pléthore de méthodes de synthèse sonore. Voici une énumération des types principaux [21].

- Synthèse sonore digitale abstraite : des sons sont générés basés sur des représentations mathématiques, n'ayant pas nécessairement une interprétation physique.
 - Synthèse additive : un son est construit comme une somme d'oscillateurs sinusoïdaux selon la théorie de Fourier.
 - Synthèse soustractive : un son est construit comme une version filtrée d'une source riche en fréquences.
 - Synthèse à tables d'ondes : des composants sinusoïdaux (ou autres) sont construits à base de valeurs lues d'un tableau.
 - Synthèse AM et FM : une forme de synthèse additive avec une variation sinusoïdale de l'amplitude (AM) ou de la fréquence (FM).
 - Autres méthodes comme la synthèse par distorsion non linéaire, la synthèse granulaire, l'addition de pulses, etc.
- Synthèse par modèle physique : peut offrir un son plus naturel mais nécessite plus de calculs.
 - Réseaux de masses ponctuelles et ressorts : une source de son est modélisée comme un grand nombre de masses ponctuelles interconnectées par des ressorts.
 - Synthèse modale : simulation de la dynamique d'un système basée sur sa décomposition modale.
 - Guides d'ondes numériques : approximation en une dimension qui peut être calculée par des lignes de retard et qui réduit les calculs nécessaires.

- Méthodes hybrides : généralisation des guides d'ondes numériques. Des ondes sont propagées et reflétées dans un réseau.
- Simulation numérique directe. Issue de l'étude acoustique, on établit les équations gouvernant un son. Elles sont résolues, souvent par des méthodes numériques.
- Autres
 - Synthèse concaténative. Des extraits de sons enregistrés sont combinés pour approximer un son cible.

Selon cette catégorisation, la méthode de synthèse sonore basée sur l'identification en source et système non linéaire est à rapprocher de la synthèse soustractive, où à la fois la source et le filtre (ici un système non linéaire) sont estimés sur des signaux audio d'un instrument.

Des objectifs de la synthèse sonore sont par exemple [21] la génération de sons pour des compositions électroacoustiques, la synthèse de parole, l'étude d'un son sans avoir besoin d'un générateur de son physique.

Chapitre 2

Outils de base

Ce chapitre présente trois outils de base qui sont utilisés dans les contributions : la méthode LPC (Section 2.1), le modèle de représentation d'état non linéaire utilisé (Section 2.2) et les séries de Volterra (Section 2.3).

2.1 Présentation de la méthode LPC

La méthode LPC a été développée pour la voix (voir [3]). Elle permet de faire une identification par trame du signal de sortie, étant donné que ce signal est le seul signal mesuré. La présentation qui en est faite ci-dessous s'appuie sur le document [16, Chapitre 11] et est réadaptée à l'identification de sons instrumentaux : une seule trame est considérée.

2.1.1 Cadre et hypothèses requises

La méthode LPC repose sur les hypothèses suivantes.

H1 Le signal y est nul en dehors du support temporel discret $\mathbb{T} \triangleq \{1, \dots, N\}$.

H2 La source échantillonnée u est représentée par un processus gaussien de moyenne nulle et d'écart type $\sigma > 0$, satisfaisant la loi de probabilité

$$p_U(u[n]|\sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{u[n]^2}{2\sigma^2}\right), \forall n \in \mathbb{T}. \quad (2.1)$$

H3 Le système à mémoire est modélisé par un filtre stationnaire tout-pôle d'ordre $2K$ (représentant K formants) décrit par la fonction de transfert

$$H(z) = \frac{1}{1 + \sum_{k=1}^{2K} a_k z^{-k}}. \quad (2.2)$$

Ces hypothèses sont regroupées comme $\mathcal{H} \triangleq \{\mathbf{H1}, \mathbf{H2}, \mathbf{H3}\}$.

Remarque En général, le numérateur de la fonction de transfert $H(z)$ dans l'Équation (2.2) est une constante de gain G . Or ce gain joue le même rôle que l'écart type de la source σ dans notre modèle; nous pouvons donc choisir $G = 1$, ce qui nous permet d'éviter que le problème soit surdéterminé.

2.1.2 Estimation des paramètres

La méthode LPC consiste à estimer les paramètres $\Theta \triangleq [\sigma \ a_1 \ \dots \ a_{2K}]^\top \triangleq [\sigma \ | \ \Theta_a]^\top$ qui maximisent la vraisemblance entre le signal y mesuré et son estimation \tilde{y} selon le modèle Θ .

D'abord, nous définissons la vraisemblance d'observer le signal y sous les hypothèses \mathcal{H} et sous l'hypothèse du modèle Θ :

$$p(y[1], \dots, y[N] | \mathcal{H}, \Theta).$$

Par le principe du maximum de vraisemblance, l'estimation des paramètres $\tilde{\Theta}$ est

$$\tilde{\Theta} \triangleq \arg \max_{\Theta} p(y[1], \dots, y[N] | \mathcal{H}, \Theta).$$

En utilisant les règles de calcul de probabilité conditionnelle et de changement de variables de façon récursive, une expression de l'anti-log-vraisemblance $L_{\mathbb{T}}$ est construite, et l'estimation des paramètres s'écrit comme

$$\tilde{\Theta} = \arg \min_{\Theta} L_{\mathbb{T}}.$$

Par le calcul détaillé en Annexe A, on peut alors prouver que les paramètres $\tilde{\Theta}$ optimaux sont la solution du système suivant [3, 16] :

$$\begin{cases} \sigma = \sqrt{\frac{Q_{\mathbb{T}}}{N}} & (2.3a) \\ R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k, \forall p \in \llbracket 1, 2K \rrbracket, & (2.3b) \end{cases}$$

où l'expression $Q_{\mathbb{T}}$ et l'autocorrélation $R_{\alpha,\beta}$ sont calculées à partir du signal y mesuré :

$$Q_{\mathbb{T}} \triangleq \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] \right)^2 \right], \quad R_{\alpha,\beta} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta], \quad (\alpha, \beta) \in \mathbb{Z}^2.$$

Remarque Par définition, l'autocorrélation est une fonction symétrique de ses arguments : $R_{\alpha,\beta} = R_{\beta,\alpha}$.

Écriture matricielle L'Équation (2.3b) peut être écrite sous la forme matricielle suivante :

$$\begin{bmatrix} R_{1,1} & R_{1,2} & \dots & R_{1,2K} \\ R_{1,2} & R_{2,2} & \ddots & R_{2,2K} \\ \vdots & \ddots & \ddots & \vdots \\ R_{1,2K} & R_{2,2K} & \dots & R_{2K,2K} \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{2K} \end{bmatrix} = \begin{bmatrix} R_{0,1} \\ R_{0,2} \\ \vdots \\ R_{0,2K} \end{bmatrix},$$

que l'on écrit matriciellement comme

$$\mathbf{G}\Theta_a = \mathbf{H}. \quad (2.4)$$

Nous concluons que le vecteur des paramètres $\tilde{\Theta}$ peut être calculé des Équations (2.4) et (2.3a).

Remarque Il est possible de faire l'approximation que $R_{i+1,j+1} \approx R_{i,j}$ en négligeant un terme dans la sommation de l'autocorrélation, du fait que $2K \ll N$. Dans ce cas, la matrice symétrique \mathbf{G} est de forme Toeplitz. Il est alors possible de réécrire les Équations (2.3) comme une équation matricielle dite « de Yule-Walker ». Dans ce cas, il existe un algorithme, l'algorithme de Durbin-Levinson, qui réduit la complexité de la LPC [16].

2.2 Modèle non linéaire en représentation d'état

La LPC repose, entre autres, sur l'hypothèse que le système dynamique peut être modélisé par un filtre tout-pôle. Dans le but de généraliser l'utilisation de la LPC à des systèmes non linéaires, nous proposons de considérer des systèmes non linéaires dont la composante linéaire est tout-pôle.

Un filtre tout-pôle pouvant être modélisé par une représentation d'état sous forme compagne [1, pp. 319-320], nous étendons cette notion en y ajoutant des termes non linéaires. Plus particulièrement, nous considérons des non-linéarités en l'état.

La représentation d'état en temps continu ($t \in \mathbb{R}$) est

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= g(\mathbf{x}(t), \mathbf{u}(t)),\end{aligned}$$

avec la sortie \mathbf{y} , et f et g deux fonctions de l'état \mathbf{x} et l'entrée \mathbf{u} . En temps discret ($n \in \mathbb{Z}$), on peut par exemple définir la représentation d'état comme étant

$$\begin{aligned}\mathbf{x}[n] &= \hat{f}(\mathbf{x}[n-1], \mathbf{u}[n]) \\ \mathbf{y}[n] &= \hat{g}(\mathbf{x}[n-1], \mathbf{u}[n]).\end{aligned}$$

Basé sur cette forme générale en temps discret, nous définissons maintenant les modèles non linéaires considérés.

2.2.1 Modèle non linéaire général

Système \mathcal{M} : forme générale Nous considérons le modèle de système \mathcal{M} dont la relation entrée-sortie est régie par la représentation d'état à temps discret sous la forme suivante :

$$\begin{cases} \mathbf{x}[n] &= \mathbf{A}\mathbf{x}[n-1] + \mathbf{B}u[n] + \mathbf{B} \sum_{\substack{(p,q) \in \mathbb{N}^2 \\ p+q \geq 2}} M_{p,q}(\underbrace{\mathbf{x}[n-1], \dots, \mathbf{x}[n-1]}_p, \underbrace{u[n], \dots, u[n]}_q) \\ y[n] &= \mathbf{C}\mathbf{x}[n], \end{cases} \quad (2.5)$$

où l'état \mathbf{x} consiste en une mémoire des $2K$ derniers échantillons de la sortie y :

$$\mathbf{x}(n) \triangleq [y[n - (2K - 1)] \quad \dots \quad y[n - 1] \quad y[n]]^\top \in \mathbb{R}^{2K \times 1}$$

et

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & 1 \\ \hline a_{2K} & \dots & \dots & \dots & a_1 \end{bmatrix} \in \mathbb{R}^{2K \times 2K},$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \in \mathbb{R}^{2K \times 1}, \quad \mathbf{C} = [0 \quad \dots \quad 0 \mid 1] \in \mathbb{R}^{1 \times 2K}.$$

Les termes $M_{p,q}$ sont des fonctions multilinéaires à sortie scalaire qui représentent les non-linéarités.

2.2.2 Modèles considérés dans ce stage

Système \mathcal{M}_1 : filtre tout-pôle Dans le cas de ce système linéaire, les fonctions multilinéaires $M_{p,q}$ sont nulles. L'Équation (2.5) peut alors se simplifier en

$$\mathbf{x}[n] = \mathbf{A}\mathbf{x}[n-1] + \mathbf{B}u[n]$$

et

$$y[n] = u[n] + \sum_{k=1}^{2K} a_k y[n-k]. \quad (2.6)$$

Remarque Pour des raisons de simplicité, nous avons choisi la représentation d'Équation (2.6). Une représentation standard est

$$y[n] = u[n] - \sum_{k=1}^{2K} \hat{a}_k y[n-k]. \quad (2.7)$$

Ces représentations sont équivalentes : le filtre tout-pôle décrit par le polynôme avec les coefficients $[\hat{a}_0, \hat{a}_1, \dots, \hat{a}_{2K}]$ devient $[1, \hat{a}_1, \dots, \hat{a}_{2K}]$ dans Éq. (2.7) et $[1, a_1 \triangleq -\hat{a}_1, \dots, a_{2K} \triangleq -\hat{a}_{2K}]$ dans Éq. (2.6).

Dans ce modèle, nous avons conservé (comme pour la voix) un nombre pair ($2K$) de paramètres (a_k) pour pouvoir encoder jusqu'à K résonances. Un nombre impair pourrait être utilisé mais contraindrait à ce qu'au moins un pôle soit réel.

Système \mathcal{M}_2 : non-linéarités quadratiques Ce modèle est similaire au modèle linéaire \mathcal{M}_1 auquel est rajoutée une non-linéarité quadratique :

$$\mathbf{x}[n+1] = \mathbf{A}\mathbf{x}[n] + \mathbf{B}u[n] + \mathbf{B}M_{2,0}(\mathbf{x}[n], \mathbf{x}[n]) \quad (2.8)$$

où $M_{2,0}(\mathbf{x}[n], \mathbf{x}[n]) \triangleq \mathbf{x}[n]^\top \mathbf{L}\mathbf{x}[n]$ avec la matrice \mathbf{L} triangulaire et définie comme suit :

$$\mathbf{L} \triangleq \begin{bmatrix} l_{2K,2K} & \dots & \dots & l_{1,2K} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & l_{2,2} & l_{1,2} \\ 0 & \dots & 0 & l_{1,1} \end{bmatrix}.$$

Notation Nous définissons également une représentation vectorielle \mathbf{l} des éléments non nuls de la matrice \mathbf{L} :

$$\mathbf{l} \triangleq [l_{1,1}, l_{1,2}, \dots, l_{2,2}, \dots, l_{2K,2K}] \in \mathbb{R}^{2K^2+K}.$$

L'Équation (2.5) peut alors se simplifier en

$$y(n) = u[n] + \sum_{k=1}^{2K} a_k y[n-k] + \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j]. \quad (2.9)$$

Dans la suite, on considère le cas où le système linéarisé ($l_{i,j} = 0$) est strictement stable, de sorte que le système (2.9) non linéarisé admet un domaine de stabilité au voisinage du point d'équilibre 0. L'étude précise du domaine de stabilité n'est pas traitée ici mais le lecteur pourra trouver dans [22] un résultat pour le cas des systèmes différentiels quadratiques adaptable au cas à temps discret.

2.3 Séries de Volterra

2.3.1 Introduction

Les séries de Volterra sont définies comme suit.

Définition 1 (Série de Volterra [6]). *Un système dynamique causal et invariant dans le temps peut être défini par la série de Volterra $\{h_n\}_{n \geq 1}$ de rayon de convergence $\rho \in \mathbb{R}_+^*$ si, pour les entrées $u(t)$ telles que $\|u(t)\| \leq \rho$, la sortie $y(t)$ donnée par*

$$y(t) = \sum_{n=1}^N \int_{\mathbb{R}^n} h_n(\tau_1, \dots, \tau_n) u(t - \tau_1) \dots u(t - \tau_n) d\tau_1 \dots d\tau_n, \forall t$$

est normalement convergente. On appelle les termes h_n les noyaux de Volterra. ■

Dans ce stage, nous supposons que l'on se trouve toujours dans la région de convergence.

Des règles pour le calcul des noyaux de Volterra d'une interconnexion de systèmes sont données dans [9, 8].

On peut aussi considérer les noyaux de transfert qui sont les transformées de Laplace multi-variables monolatérales des noyaux de Volterra. En général, les noyaux de transfert ne sont pas uniques mais il existe des formes uniques, comme les formes symétrique et triangulaire [6].

Les séries de Volterra sont entre autres utilisées pour la modélisation et simulation des instruments de musique, des circuits électroniques et du haut-parleur [9, 23, 24, 10, 8, 7].

2.3.2 Lien avec la représentation d'état

Considérons un système qui satisfait les trois conditions suivantes.

- La non-linéarité du système est régulière autour d'un point d'équilibre et l'état considéré reste proche de ce point.
- Le système a une mémoire à horizon fini.
- Nous considérons des états dans la région de convergence d'une série de Volterra.

Dans ce cas, on peut prouver que le système considéré peut être représenté par une série de Volterra.

Dans ce cadre, certains phénomènes comme l'hystérésis ou le chaos ne peuvent être représentés.

2.3.3 Noyaux de transfert

Dans la Figure 2.1, une réalisation des noyaux de Volterra des trois premiers ordres d'un système \mathcal{M} est montrée.

Avec la fonction de transfert du filtre \mathbf{W} égale à

$$\mathbf{W}(s) \triangleq [s\mathbf{I} - \mathbf{A}]^{-1},$$

on peut calculer que les noyaux de transfert d'ordres 1 et 2 s'expriment comme

$$H_1(s_1) \triangleq \mathbf{C}\mathbf{W}(s_1)\mathbf{B} \tag{2.10}$$

et

$$H_2(s_1, s_2) \triangleq \mathbf{C}\mathbf{W}(s_1 + s_2)M_{2,0}(\mathbf{W}(s_1)\mathbf{B}, \mathbf{W}(s_2)\mathbf{B}). \tag{2.11}$$

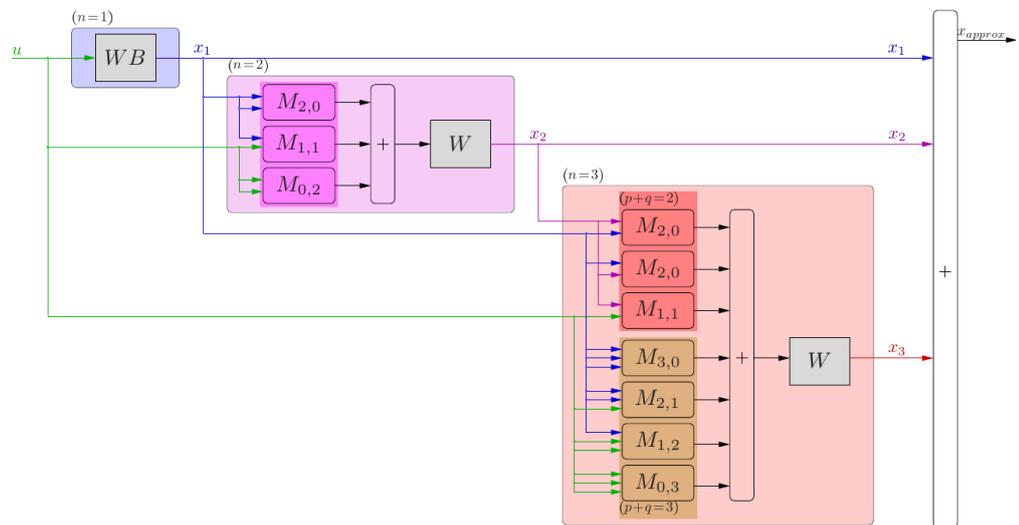


FIGURE 2.1 – Réalisation des noyaux de Volterra des trois premiers ordres [8]

Chapitre 3

Extension de la méthode LPC à un modèle quadratique (QPC)

Les travaux effectués dans le présent chapitre commencent par le développement d'une méthode d'identification du Système \mathcal{M}_2 basé sur la LPC (Section 3.1). Ensuite, la méthode est évaluée en identifiant des données artificielles dans la Section 3.2 et les résultats sont discutés dans la Section 3.3.

3.1 Présentation de la méthode QPC

Nous présentons ici la méthode d'identification du Système \mathcal{M}_2 basé sur la connaissance de sa sortie. Comme dans la méthode LPC, présentée dans la Section 2.1, nous considérons pas des signaux de parole mais des sons instrumentaux qui consistent en une trame. La seule différence est le modèle du système considéré, comme introduit ci-dessous.

3.1.1 Cadre et hypothèses requises

Parmi les Hypothèses **H1-H2-H3** requises pour l'utilisation de la LPC, l'hypothèse **H3** doit être remplacée dans notre cas par

H3' Le système à mémoire est modélisé par le Modèle \mathcal{M}_2 décrit par l'Équation (2.9).

Ces hypothèses sont regroupées comme $\mathcal{H}' \triangleq \{\mathbf{H1}, \mathbf{H2}, \mathbf{H3}'\}$.

3.1.2 Estimation des paramètres

Dans ce cas, on veut estimer les paramètres $\Xi \triangleq [\sigma \mid a_1 \dots a_{2K} \mid l_{1,1} \dots l_{2K,2K}]^\top \triangleq [\sigma \mid \Xi_a \mid \Xi_l]^\top \triangleq [\sigma \mid \Xi_{a,l}]^\top$.

Nous procédons de façon analogue au développement de la méthode LPC. Nous obtenons une estimation des paramètres $\hat{\Xi}$ par le maximum de vraisemblance :

$$\hat{\Xi} \triangleq \arg \max_{\Xi} p(y[1], \dots, y[N] | \mathcal{H}', \Xi),$$

ce qui est équivalent au minimum de l'anti-log-vraisemblance $L'_{\mathbb{T}}$:

$$\hat{\Xi} = \arg \min_{\Xi} L'_{\mathbb{T}}.$$

Par le calcul détaillé en Annexe B, on peut prouver que les paramètres $\tilde{\Xi}$ optimaux sont la solution du système suivant :

$$\left\{ \begin{array}{l} \sigma = \sqrt{\frac{Q'_T}{N}} \\ R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \\ S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j}, \end{array} \right. \quad \begin{array}{l} (3.1a) \\ \forall p \in \llbracket 1, 2K \rrbracket \\ \forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket, \end{array} \quad \begin{array}{l} (3.1b) \\ (3.1c) \end{array}$$

où l'expression Q'_T , l'autocorrélation $R_{\alpha,\beta}$ ainsi que les « autocorrélations » d'ordres supérieurs $S_{\alpha,\beta,\gamma}$ et $T_{\alpha,\beta,\gamma,\delta}$ sont calculées à partir du signal y mesuré :

$$Q'_T \triangleq \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right)^2 \right],$$

$$\left\{ \begin{array}{l} R_{\alpha,\beta} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta] \\ S_{\alpha,\beta,\gamma} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta] y[n-\gamma] \\ T_{\alpha,\beta,\gamma,\delta} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta] y[n-\gamma] y[n-\delta], \end{array} \right.$$

avec $(\alpha, \beta, \gamma, \delta) \in \mathbb{Z}^4$.

Nous appelons la méthode décrite *Codage Prédictif Quadratique* (QPC, de l'anglais *Quadratic Predictive Coding*).

Remarques Les autocorrélations (d'ordres supérieurs) R , S et T sont des fonctions symétriques de leurs arguments, que l'on peut permuter.

Des approximations comme $R_{i+1,j+1} \approx R_{i,j}$ sont possibles, où la valeur de $R_{\alpha,\beta}$ ne dépend que de $|\alpha - \beta|$, et analogue pour S et T . Afin d'éviter toute influence numérique, nous décidons de garder les valeurs exactes.

Écriture matricielle Les Équations (3.1b)-(3.1c) peuvent être écrites sous la forme matricielle suivante :

$$\left[\begin{array}{ccc|ccc} R_{1,1} & \dots & R_{1,2K} & S_{1,1,1} & \dots & S_{1,2K,2K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ R_{2K,1} & \dots & R_{2K,2K} & S_{2K,1,1} & \dots & S_{2K,2K,2K} \\ \hline S_{1,1,1} & \dots & S_{2K,1,1} & T_{1,1,1,1} & \dots & T_{1,1,2K,2K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ S_{1,2K,2K} & \dots & S_{2K,2K,2K} & T_{2K,2K,1,1} & \dots & T_{2K,2K,2K,2K} \end{array} \right] \cdot \begin{bmatrix} a_1 \\ \vdots \\ a_{2K} \\ l_{1,1} \\ \vdots \\ l_{2K,2K} \end{bmatrix} = \begin{bmatrix} R_{0,1} \\ \vdots \\ R_{0,2K} \\ S_{0,1,1} \\ \vdots \\ S_{0,2K,2K} \end{bmatrix},$$

que l'on écrit matriciellement comme

$$\left[\begin{array}{c|c} \mathbf{G}'_R & \mathbf{G}'_S \\ \mathbf{G}'_T & \mathbf{G}'_T \end{array} \right] \cdot \begin{bmatrix} \mathbf{a} \\ \mathbf{l} \end{bmatrix} = \begin{bmatrix} \mathbf{H}'_R \\ \mathbf{H}'_S \end{bmatrix},$$

ou encore

$$\mathbf{G}'\boldsymbol{\Xi}_{\mathbf{a},\mathbf{l}} = \mathbf{H}'. \quad (3.2)$$

L'Équation (3.2) est linéaire en les paramètres.

Nous concluons que $\tilde{\boldsymbol{\Xi}}$ peut être calculé des Équations (3.2) et (3.1a).

Implémentation pratique Nous définissons le vecteur

$$\bar{\mathbf{y}} \triangleq [y(2), \dots, y(N)]^\top \in \mathbb{R}^{(N-1) \times 1}$$

et les matrices

$$\bar{\mathbf{Y}}_1 \triangleq \begin{bmatrix} y[1] & 0 & \dots & 0 \\ y[2] & y[1] & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ y[N-1] & y[N-2] & \dots & y[N-2K] \end{bmatrix} \in \mathbb{R}^{(N-1) \times 2K},$$

qui contient des versions retardées de parties de y , correspondant au vecteur \mathbf{a} , et

$$\bar{\mathbf{Y}}_2 \triangleq \begin{bmatrix} y^2[1] & 0 & \dots & 0 & \dots & \dots & 0 \\ y^2[2] & y[1]y[2] & \dots & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & \dots & \dots & \vdots \\ y^2[N-1] & y[N-1]y[N-2] & \dots & y^2[N-2] & \dots & \dots & y^2[N-2K] \end{bmatrix} \in \mathbb{R}^{(N-1) \times (2K^2+K)},$$

qui contient des versions retardées de carrés de parties de y , correspondantes au vecteur \mathbf{l} .

On peut alors obtenir \mathbf{G}' et \mathbf{H}' par les expressions suivantes :

$$\begin{cases} \mathbf{G}'_R = \bar{\mathbf{Y}}_1^\top \bar{\mathbf{Y}}_1 \\ \mathbf{G}'_S = \bar{\mathbf{Y}}_1^\top \bar{\mathbf{Y}}_2 \\ \mathbf{G}'_T = \bar{\mathbf{Y}}_2^\top \bar{\mathbf{Y}}_2 \\ \mathbf{H}'_R = \bar{\mathbf{Y}}_1^\top \bar{\mathbf{y}} \\ \mathbf{H}'_S = \bar{\mathbf{Y}}_2^\top \bar{\mathbf{y}}. \end{cases} \quad (3.3)$$

Ces matrices sont utilisées dans le calcul des vecteurs \mathbf{a} et \mathbf{l} . Quant à l'estimation de l'écart type, nous pouvons la calculer comme

$$\tilde{\sigma} = \sqrt{\frac{Q'_T}{N}} = \frac{\|\bar{\mathbf{y}} - \bar{\mathbf{Y}}_1 \tilde{\mathbf{a}} - \bar{\mathbf{Y}}_2 \tilde{\mathbf{l}}\|_2}{\sqrt{N-1}}. \quad (3.4)$$

3.2 Évaluation de la méthode

Dans cette section, nous évaluons la méthode en identifiant des données artificielles. Dans la Sous-section 3.2.1, le protocole de ce test est développé : la génération des données utilisées, l'implémentation de la méthode QPC et la définition des erreurs qui sont utilisées pour l'évaluation dans la Sous-section 3.2.2.

3.2.1 Protocole

Génération des données

Afin de tester la méthode QPC, on simule numériquement un système qui suit le modèle \mathcal{M}_2 . Les paramètres utilisés sont les suivants :

- l'entrée est un processus gaussien de moyenne nulle et d'écart type $\sigma \in \{0,05; 0,1; 0,5; 1\}$;
- plusieurs longueurs N de signaux sont considérées : $N \in \{10^3, 10^4, 10^5, 10^7\}$, ce qui correspond à des durées de $\{0,023\text{s}, 0,23\text{s}, 2,3\text{s}, 3,8\text{min}\}$ pour une fréquence d'échantillonnage de 44100Hz ;
- le système est d'ordre $2K = 8$;
- le vecteur

$$\mathbf{a} = [1 \quad 0,1824 \quad -0,0166 \quad 0,0492 \quad -0,0088 \quad 0,0315 \quad -0,0068 \quad 0,0478 \quad -0,1678] ;$$

- le vecteur

$$\mathbf{l} = \begin{bmatrix} -0,0695 & -0,1 & 0,072 & -0,0518 & 0,0373 & -0,0268 & 0,0193 & -0,0139 \\ 0,05 & -0,0518 & 0,0373 & -0,0268 & 0,0193 & -0,0139 & 0,01 & -0,0072 \\ -0,036 & -0,0268 & 0,0193 & -0,0139 & 0,01 & -0,0072 & 0,0052 & -0,0037 \\ 0,0259 & -0,0139 & 0,01 & -0,0072 & 0,0052 & -0,0037 & 0,0027 & -0,0019 \\ -0,0186 & -0,0072 & 0,0052 & -0,0037 & 0,0027 & -0,0019 & 0,0069 & 0,0069 \\ 0,0134 & -0,0037 & 0,0027 & -0,0019 & 0,0069 & 0,0069 & 0,0069 & 0,0069 \\ -0,0097 & -0,0019 & 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 \\ 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 & 0,0069 \end{bmatrix}.$$

Une représentation graphique des vecteurs \mathbf{a} et \mathbf{l} est donnée dans la Figure 3.1.

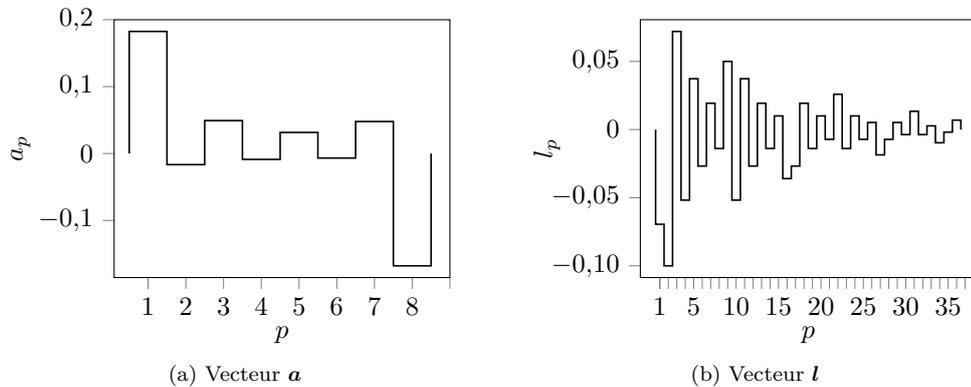


FIGURE 3.1 – Représentation des vecteurs \mathbf{a} et \mathbf{l} du système simulé.

Remarque L'écart type σ de la source devrait être choisi en fonction de la région de convergence de la série de Volterra ; il influence la stabilité de la simulation. Une remarque est faite en Annexe D. Dans ce stage, nous ne considérons pas la région de convergence. Il est alors possible qu'une simulation avec certaines valeurs de σ et N soit parfois ou toujours instable.

Implémentation numérique de la méthode

La méthode QPC est implémentée en PYTHON en utilisant les Équations (3.3)-(3.4).

Nous mentionnons que l'Hypothèse **H1** n'est pas satisfaite pour quelques échantillons après $y[N]$. Nous avons vérifié que l'effet numérique de ce fait est négligeable.

Définition des erreurs

Pour l'évaluation de la performance des méthodes développées, l'erreur e entre un vecteur estimé $\tilde{\mathbf{b}}$ et le vecteur de référence \mathbf{b} est définie comme suit (où la référence n'est pas nécessairement mentionnée explicitement) :

$$e(\tilde{\mathbf{b}}) \triangleq e(\tilde{\mathbf{b}}, \mathbf{b}) \triangleq 20 \log \frac{\|\tilde{\mathbf{b}} - \mathbf{b}\|_p}{\|\mathbf{b}\|_p} \text{ [dB]},$$

avec la norme p définie par

$$\|\mathbf{b}\|_p \triangleq \left(\sum_{i=1}^n |b_i|^p \right)^{\frac{1}{p}}.$$

Dans la suite, nous considérons la norme euclidienne ($p = 2$).

3.2.2 Résultats

Un premier test est d'identifier le système pour un écart type de la source constant qui vaut $\sigma = 0,5$ en changeant le nombre de données $N \in \{10^3, 10^5, 10^7\}$. Une représentation graphique de l'évolution des estimations est donnée en Figure 3.2, avec les erreurs correspondantes en Tableau 3.1.

Nous observons que pour un nombre croissant de données, les estimations s'améliorent. Cependant, l'estimation du vecteur \mathbf{l} est toujours considérablement moins bonne que celle de la matrice \mathbf{a} .

N	$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)
10^3	-49,8	-11,2	4,4
10^5	-50,6	-29,2	-17,6
10^7	-93,8	-51,4	-34,2

TABLEAU 3.1 – Estimation du système dans le cas $\sigma = 0,5$: évolution des erreurs en fonction du nombre de données N .

Un deuxième test consiste à varier l'écart type de la source σ pour un nombre de données N donné. L'influence sur l'erreur est montrée dans le Tableau 3.2.

Ici, on remarque que l'erreur diminue souvent en augmentant l'écart type σ . Bien que le tableau ne contienne que des valeurs d'une seule évaluation et non pas des variance des valeurs, on peut supposer que le choix de σ influence l'importance relative des contributions linéaire et quadratique du système dans le signal y . Ce fait n'a pas été étudié plus profondément.

Nous concluons que toutes les estimations (de l'écart type σ , des vecteurs \mathbf{a} et \mathbf{l}) s'améliorent lorsque N augmente. Les estimations de σ et des a_k sont correctes, même pour un signal court. Par contre, il est nécessaire d'avoir un nombre d'échantillons très important pour estimer correctement le vecteur \mathbf{l} .

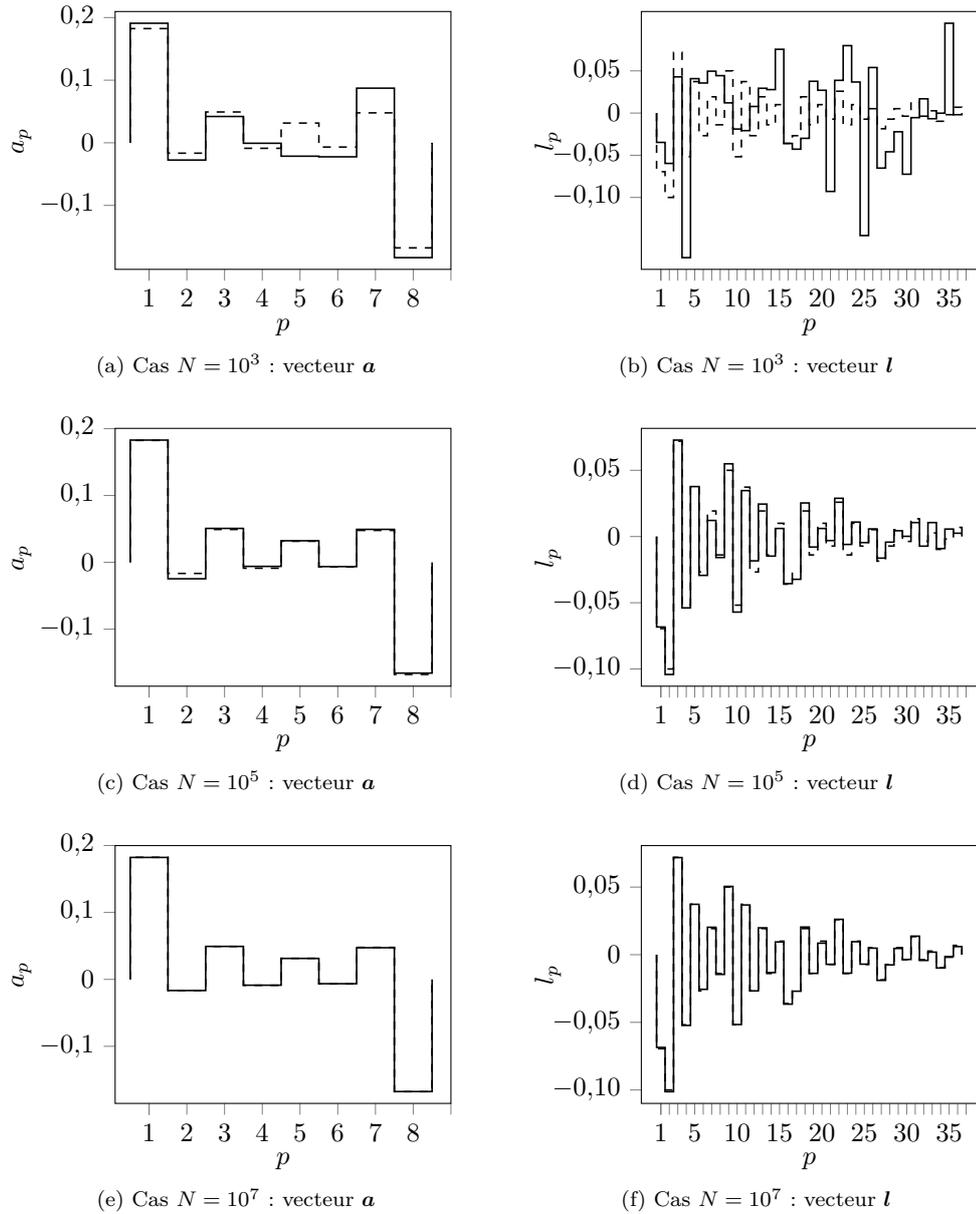


FIGURE 3.2 – Estimation du système dans le cas $\sigma = 0,5$: représentation graphique des estimations des vecteurs \mathbf{a} et \mathbf{l} . Les lignes tiretées indiquent les valeurs exactes \mathbf{a} et \mathbf{l} , les lignes pleines leurs estimations $\tilde{\mathbf{a}}$ et $\tilde{\mathbf{l}}$.

3.3 Discussion

Dans la Sous-section 3.2.2, nous avons constaté que la méthode QPC fonctionne mais nécessite un nombre élevé de données. Dans notre cas, qui est l'application à la synthèse sonore, nous ne disposons pas de tant de données. Par exemple, un son d'une durée de 5s à une fréquence

σ	N	$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)
0,05	10^3	-34,2	-9,2	26,5
0,05	10^4	-43,6	-16,1	16,2
0,05	10^5	-52,4	-32,5	3,9
0,1	10^3	-35,4	-11,9	19,2
0,1	10^4	-48,3	-17,7	10,6
0,1	10^5	-57,0	-31,1	-0,9
0,5	10^3	-31,4	-11,6	4,9
0,5	10^4	-66,5	-18,4	-3,0
0,5	10^5	-56,5	-30,3	-13,9
1	10^3	-40,2	-11,6	-1,3
1	10^4	-46,1	-19,2	-12,7
1	10^5	-49,7	-26,9	-20,2

TABLEAU 3.2 – Estimation du système : évolution des erreurs en fonction de l'écart type σ et du nombre de données N .

d'échantillonnage de 44100Hz correspond à $N = 2,205 \cdot 10^5$ échantillons.

Dans un premier temps, nous avons considéré plusieurs pistes dont notamment l'analyse de la variance des estimateurs présents dans les matrices \mathbf{G}' et \mathbf{H}' . Dans ce contexte, nous avons entre autres abordé l'information de Fisher avec la borne de Cramer-Rao, et l'analyse par intervalle. Plus d'informations sur ces réflexions sont mentionnées en Annexe C. Nous avons abandonné ces méthodes parce qu'elles présentaient des calculs longs et/ou ne semblaient pas appropriées pour notre problème.

Finalement, nous avons choisi d'étudier la régularisation dans le but d'améliorer la précision de l'estimation du vecteur \mathbf{l} en utilisant un nombre limité de données. La régularisation est le sujet du chapitre suivant.

Chapitre 4

Régularisation de l'estimation

Dans ce chapitre, nous rajoutons une régularisation à la méthode d'identification QPC dans le but d'obtenir une estimation plus précise des paramètres du système. Dans la Section 4.1, nous expliquons le principe de la régularisation et mentionnons les travaux existants pour le cas de l'identification d'un système linéaire. Ensuite, nous développons deux extensions de la méthode QPC, pour l'identification du système \mathcal{M}_2 , où nous rajoutons une régularisation sur les noyaux de Volterra du premier ordre (Section 4.2) et des premier et deuxième ordres (Section 4.3). Nous appelons l'ensemble de ces deux méthodes Codage Prédicatif Quadratique Régularisé (RQPC, de l'anglais *Regularized Quadratic Predictive Coding*).

4.1 Introduction

Pour l'application à la synthèse sonore, nous ne disposent que d'un nombre limité de données et nous voulons utiliser le maximum de l'information présente dans ces données [18]. De plus, le fait d'améliorer les estimations est également important en vue du rajout d'ordres supérieurs dans le modèle. Dans le but d'améliorer l'estimation des paramètres du modèle \mathcal{M}_2 , nous proposons de rajouter une régularisation à la méthode QPC.

Nous commençons cette section introductive par le principe général de la régularisation (Sous-section 4.1.1). En partant d'une méthode existante pour le cas linéaire, où le spectre est régularisé (Sous-section 4.1.2), nous proposons une extension au cas quadratique où les noyaux de transfert de Volterra des ordres 1 et 2 sont régularisés (Sous-section 4.1.3). Finalement, dans la Sous-section 4.1.4, nous expliquons comment il faut estimer les valeurs des hyperparamètres qui seront introduits.

4.1.1 Principe général de la régularisation

Le principe général de la régularisation est le fait de prendre en compte une probabilité a priori sur les paramètres du modèle que l'on estime [25, 20]. Cette distribution introduit une certaine régularité de l'estimation et évite ainsi de surmodéliser les données.

Afin de pouvoir inclure cette distribution a priori sur les paramètres dans la méthode d'identification, nous changeons le point de vue utilisé dans la QPC. Au lieu de maximiser la vraisemblance d'observer le signal de sortie y sous l'hypothèse du modèle avec des paramètres donnés, nous maximisons la vraisemblance d'avoir un modèle avec des paramètres donnés sous l'hypothèse d'observer le signal y . Ce changement de paradigme permet de rajouter la distribution a priori en utilisant la règle de Bayes.

Dans la distribution a priori sur les paramètres, un ou plusieurs hyperparamètres sont introduits. Ces hyperparamètres balancent le poids relatif entre la probabilité a priori et la probabilité des données dans la fonction de coût. Les valeurs de ces paramètres ne sont pas choisies arbitrairement, mais doivent être déduites des données par une méthode appropriée [20].

4.1.2 Méthode existante pour le cas d'un système linéaire : lisseur du spectre

Dans [19, 20], on propose de rajouter une distribution a priori sur les paramètres d'un système linéaire que l'on estime, de sorte que l'on impose un degré de lisseur (*smoothness* en anglais) au spectre du filtre linéaire.

Le spectre du filtre linéaire contient une expression

$$P(f) \triangleq \sum_{k=1}^{2K} a_k e^{j2\pi f k}.$$

Un « facteur de lissage linéaire d'ordre k » est défini comme

$$R_k^{(1)} \triangleq \int_0^1 \left| \frac{\partial^k}{\partial f^k} P(f) \right|^2 df, \text{ avec } k \in \mathbb{N}^*.$$

Ce facteur est grand pour un spectre qui est peu lisse : la dérivée k -ième atteint des valeurs importantes, dont le carré est intégré par rapport à la fréquence. Un facteur plus petit correspond à un spectre plus lisse, et dans le cas limite d'un facteur qui vaut nul, on a affaire à une fonction constante. La signification du facteur $R_k^{(1)}$ est illustrée dans la Figure 4.1. Dans [20], on mentionne que le choix de l'ordre k a peu d'influence et la valeur $k = 1$ est choisie.

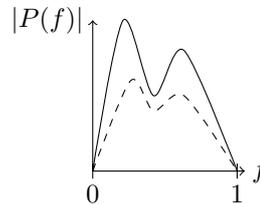


FIGURE 4.1 – Signification du facteur $R_k^{(1)}$: le spectre en ligne pleine est moins lisse que le spectre en ligne tiretée donc son facteur $R_k^{(1)}$ est plus grand.

Ce facteur de lissage est utilisé dans la définition d'une distribution a priori sur les paramètres linéaires \mathbf{a} qui contient aussi un hyperparamètre λ_1 . Une bonne valeur de cet hyperparamètre fait en sorte que le lisseur imposé correspond à une meilleure estimation des paramètres linéaires. Une méthode d'estimation de λ_1 est mentionnée ci-dessous.

4.1.3 Extension au modèle quadratique : lisseur des noyaux de transfert de Volterra

Le fait de limiter la variation du spectre du filtre linéaire dans le cas de l'estimation des paramètres d'un système linéaire a une interprétation physique et mène à une méthode d'estimation pratique. Il n'est pas facile d'étendre cette notation à des ordres supérieurs. De l'observation que

le fait de limiter la variation du filtre linéaire est équivalent au fait de limiter la variation du noyau de transfert de Volterra du premier ordre, nous proposons la régularisation des noyaux de transfert de Volterra comme régularisation d'ordres supérieurs.

4.1.4 Calcul des hyperparamètres

La méthode régularisée pour un système linéaire contient deux hyperparamètres : la valeur de l'hyperparamètre λ_1 et l'ordre k . Dans le cas de la régularisation de plusieurs noyaux de transfert de Volterra, on a affaire à plusieurs hyperparamètres λ_i et ordres respectifs k_i .

Puisque l'on mentionne dans [20] que la valeur de k est moins importante que la valeur de λ_1 , nous proposons de choisir les ordres k_i arbitrairement, et de développer une méthode d'estimation pour les paramètres λ_i .

Il existe plusieurs méthodes pour le calcul des hyperparamètres λ_i (voir [20] pour des méthodes dans le cas d'un système linéaire). Nous proposons d'utiliser une extension au modèle \mathcal{M}_2 de la méthode de la Validation Croisée Généralisée (GCV, de l'anglais *Generalized Cross Validation*, [26]). Dans cette méthode, les valeurs des hyperparamètres sont trouvées en minimisant la différence entre la sortie mesurée et la sortie prédite par le modèle estimé en fonction de ces hyperparamètres. Nous n'établissons pas une expression formelle pour ce minimum, mais le déterminons numériquement comme dans [20].

Nous précisons que dans [20], l'écart type de la source est vue comme un hyperparamètre. Cependant, dans nos méthodes, nous le calculons par le principe du maximum de vraisemblance d'une fonction de coût, comme nous préciserons ci-dessous.

4.2 Méthode avec régularisation du noyau d'ordre 1 (R₁QPC)

4.2.1 Présentation de la méthode

Nous présentons ici les étapes du développement de la méthode appelée R₁QPC (Codage Prédicatif Quadratique Régularisé, de l'anglais *Regularized Quadratic Predictive Coding*), où l'indice 1 précise que l'on régularise le noyau de transfert de Volterra d'ordre 1. Les calculs détaillés correspondants à la présente Sous-section se trouvent dans la Section E.1.

1. Calcul du noyau de transfert de Volterra d'ordre 1

On calcule que le noyau de transfert de Volterra d'ordre 1 correspond à

$$H_1(z) = \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)}. \quad (4.1)$$

2. A priori de lisseur

À partir du noyau de transfert de Volterra d'ordre 1 (Équation (4.1)), nous définissons le spectre suivant que nous voulons régulariser :

$$P(f) \triangleq \sum_{p=1}^{2K} a_p e^{j2\pi fp}.$$

Pour cela, on définit le terme de « facteur de lissage linéaire d'ordre k » $R_k^{(1)}$ comme suit :

$$R_k^{(1)} \triangleq \int_0^1 \left| \frac{d^k}{df^k} P(f) \right|^2 df.$$

Avec la définition de la matrice

$$\Delta_k \triangleq \begin{bmatrix} 1^{2k} & 0 & 0 & \dots & 0 \\ 0 & 2^{2k} & 0 & \dots & 0 \\ 0 & 0 & 3^{2k} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & (2K)^{2k} \end{bmatrix},$$

nous obtenons

$$R_k^{(1)} = (-4\pi^2)^k \mathbf{a}^\top \Delta_k \mathbf{a}.$$

3. Expression de la vraisemblance

Afin de pouvoir inclure une distribution a priori sur les paramètres dans la fonction de coût, nous ne voulons pas maximiser la vraisemblance P'_N d'observer le signal y sous l'hypothèse du modèle Ξ (ce qui était le cas dans la méthode QPC, cf. Annexe B), mais nous voulons maximiser la vraisemblance M_{N,R_1} d'avoir un modèle avec les paramètres Ξ_{R_1} sous l'hypothèse d'observer le signal y :

$$\tilde{\Xi}_{R_1} \triangleq \arg \max_{\Xi_{R_1}} M_{N,R_1}, \text{ avec } M_{N,R_1} \triangleq p(\Xi_{R_1} | y, \mathcal{H}').$$

Nous utilisons la règle de Bayes, et introduisons une distribution a priori des paramètres \mathbf{a} de la forme

$$p(\Xi_{R_1}) = (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_a)^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{a}^\top \mathbf{R}_a^{-1} \mathbf{a}} \quad (4.2)$$

avec la matrice de corrélation \mathbf{R}_a qui contient l'information sur le lisseur et qui correspond à

$$\mathbf{R}_a^{-1} = \frac{\lambda_1}{\sigma^2} \Delta_k, \quad (4.3)$$

comme dans [20].

Nous obtenons l'expression suivante pour la vraisemblance :

$$P'_N \cdot p(\Xi_{R_1}) = (2\pi)^{-N} \frac{\lambda_1^K}{\sigma^{2K+N}} [(2K)!]^k e^{-\frac{1}{2\sigma^2} (Q'_T + \lambda_1 \mathbf{a}^\top \Delta_k \mathbf{a})}.$$

Nous définissons

$$Q_{T,R_1} \triangleq Q'_T + \lambda_1 \mathbf{a}^\top \Delta_k \mathbf{a}.$$

4. Maximisation de la vraisemblance

On peut prouver que les paramètres $\tilde{\Xi}_{R_1}$ sont la solution du système suivant :

$$\left\{ \begin{array}{l} \sigma = \sqrt{\frac{Q_{T,R_1}}{N + 2K}} \end{array} \right. \quad (4.4a)$$

$$\left\{ \begin{array}{l} R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k - \lambda_1 \cdot p^{2k} a_p + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \end{array} \right. \quad \forall p \in \llbracket 1, 2K \rrbracket \quad (4.4b)$$

$$\left\{ \begin{array}{l} S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j}, \end{array} \right. \quad \forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket. \quad (4.4c)$$

Écriture matricielle Les Équations (4.4) peuvent être écrites sous la forme matricielle suivante :

$$\left(\begin{array}{ccc|ccc} R_{1,1} & \dots & R_{1,2K} & S_{1,1,1} & \dots & S_{1,2K,2K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ R_{2K,1} & \dots & R_{2K,2K} & S_{2K,1,1} & \dots & S_{2K,2K,2K} \\ \hline S_{1,1,1} & \dots & S_{2K,1,1} & T_{1,1,1,1} & \dots & T_{1,1,2K,2K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ S_{1,2K,2K} & \dots & S_{2K,2K,2K} & T_{2K,2K,1,1} & \dots & T_{2K,2K,2K,2K} \end{array} \right) - \lambda_1 \left(\begin{array}{cc|cc} 1^{2k} & 0 & & \\ & \ddots & & \\ 0 & & (2K)^{2k} & 0 \\ \hline & & 0 & 0 \end{array} \right) \cdot \begin{bmatrix} a_1 \\ \vdots \\ a_{2K} \\ l_{1,1} \\ \vdots \\ l_{2K,2K} \end{bmatrix} = \begin{bmatrix} R_{0,1} \\ \vdots \\ R_{0,2K} \\ S_{0,1,1} \\ \vdots \\ S_{0,2K,2K} \end{bmatrix},$$

que l'on écrit matriciellement comme

$$\left(\begin{array}{c|c} \mathbf{G}_{R,R_1} & \mathbf{G}_{S,R_1} \\ \hline \mathbf{G}_{S,R_1}^T & \mathbf{G}_{T,R_1} \end{array} \right) - \lambda_1 \underbrace{\begin{bmatrix} \Delta_k & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix}}_{\triangleq D_1} \cdot \begin{bmatrix} \mathbf{a} \\ l \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{R,R_1} \\ \mathbf{H}_{S,R_1} \end{bmatrix},$$

ou encore

$$(\mathbf{G}_{R_1} - \lambda_1 \mathbf{D}_1) \Xi_{\mathbf{a},l,R_1} = \mathbf{H}_{R_1},$$

abrégée comme

$$\mathbf{G}_{\lambda,R_1} \Xi_{\mathbf{a},l,R_1} = \mathbf{H}_{R_1}. \quad (4.5)$$

L'Équation (4.5) est linéaire en les paramètres.

Nous concluons que $\tilde{\Xi}_{R_1}$ peut être calculé des Équations (4.5) et (4.4a).

Implémentation pratique Comme dans le cas de la méthode QPC, on utilise le vecteur $\bar{\mathbf{y}}$ et les matrices $\bar{\mathbf{Y}}_1$ et $\bar{\mathbf{Y}}_2$ (définis dans la Sous-section 3.1.2) pour l'implémentation pratique.

5. Estimation de l'hyperparamètre λ_1

Il nous reste à estimer la valeur $\tilde{\lambda}_1$ de l'hyperparamètre λ_1 , présent dans les formules de la méthode R₁QPC. Pour le faire, nous adaptons la méthode GCV, utilisée dans [20, 26] dans le cas d'un système linéaire, à notre cas. L'hyperparamètre est trouvé en minimisant une expression dite GCV :

$$\tilde{\lambda}_1 = \arg \min \text{GCV}(\lambda_1).$$

Cette expression est définie comme

$$\text{GCV}(\lambda_1) = \frac{\|(\mathbf{I} - \mathbf{M})\bar{\mathbf{y}}\|_2}{\text{Tr}(\mathbf{I} - \mathbf{M})}, \quad (4.6)$$

avec (cf. notation introduite en Sous-section 3.1.2)

$$\mathbf{M} \triangleq \begin{bmatrix} \bar{\mathbf{Y}}_1 & | & \bar{\mathbf{Y}}_2 \end{bmatrix} \cdot (\mathbf{G}_{R_1} - \lambda_1 \mathbf{D}_1)^{-1} \cdot \begin{bmatrix} \bar{\mathbf{Y}}_1^\top \\ \bar{\mathbf{Y}}_2^\top \end{bmatrix}.$$

On note que le numérateur de la GCV exprime la différence entre le signal de sortie y mesuré et la sortie prédite par le modèle obtenu par la méthode R₁QPC.

Implémentation pratique On observe que le calcul en PYTHON de \mathbf{M} mène à des problèmes de mémoire à cause de la taille des matrices. Une reformulation de la GCV est proposée en annexe, qui résout ce problème.

4.2.2 Évaluation

Pour l'évaluation de la méthode R₁QPC, le même système que pour l'évaluation de la GCV est simulé.

Avec valeur de λ_1 obtenue par la méthode GCV Nous implémentons la méthode R₁QPC en PYTHON et choisissons $k = 1$. Nous approximons le minimum de la GCV numériquement en calculant l'expression pour beaucoup de valeurs de λ_1 . Il se trouve que l'on obtient des valeurs négatives de $\tilde{\lambda}_1$, comme par exemple dans le cas illustré dans la Figure 4.2.

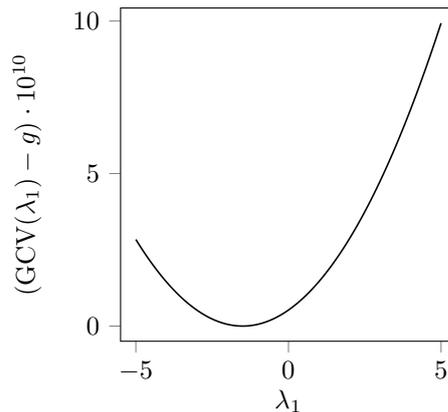


FIGURE 4.2 – Estimation $\tilde{\lambda}_1$ pour une simulation avec $N = 10^5$, $\sigma = 1$. Pour des raisons numériques, dans la figure, la valeur de la GCV est normalisée en soustrayant le minimum g atteint sur l'intervalle considéré. Le minimum g de la GCV est obtenu pour $\tilde{\lambda}_1 = -1,500$.

Par la définition de la matrice \mathbf{R}_a , la valeur de λ_1 doit être positive. Nous utilisons quand-même la valeur obtenue pour comparer les méthodes QPC et R₁QPC appliquées au signal de sortie issue du même signal d'entrée avec $N = 10^5$, $\sigma = 1$, qui était utilisée pour le calcul de la Figure 4.2. Le Tableau 4.1 montre les erreurs obtenues.

Nous observons que les différences entre les erreurs de la méthode QPC et la méthode R₁QPC sont petites. De plus, une répétition de ce test pour des combinaisons de valeurs choisies de N et σ nous montre que les petites différences entre les erreurs sont parfois positives et parfois négatives.

Nous concluons que la méthode R₁QPC, développée et implémentée comme discutée ici, ne fonctionne pas. À l'heure actuelle, nous n'avons pu trouver la raison de ce fait.

TABLEAU 4.1 – Erreurs sur l'estimation des paramètres par les méthodes QPC et R₁QPC pour un même signal d'entrée avec $N = 10^5$, $\sigma = 1$, correspondant à la Figure 4.2.

QPC			R ₁ QPC		
$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)	$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)
-52,46	-29,99	-20,20	-52,26	-30,17	-20,19

Avec valeur de λ_1 choisie arbitrairement L'estimation de l'hyperparamètre λ_1 ne fonctionne pas à cette heure (un problème probablement d'implantation conduit à des λ_1 négatifs). Afin de visualiser son effet lorsqu'il est positif, nous montrons en Tableau 4.2 les résultats pour un test avec plusieurs valeurs de λ_1 positives réglées à la main. On observe bien l'effet sur l'erreur d'estimation des vecteurs \mathbf{a} et \mathbf{l} lorsque λ_1 croît de 0 à ± 70 . L'erreur sur le vecteur \mathbf{a} peut être diminuée de 3,36dB, tandis que l'erreur sur le vecteur \mathbf{l} ne peut être diminuée que de 0,06dB. On note aussi que la régularisation n'améliore pas (voire dégrade) l'estimation de σ , puisque les méthodes RQPC ne contiennent pas de régularisation sur ce paramètre.

TABLEAU 4.2 – Erreurs sur l'estimation des paramètres par les méthodes QPC et R₁QPC (en fonction de valeurs positives de λ_1) pour un même signal d'entrée avec $N = 10^5$, $\sigma = 1$.

QPC			R ₁ QPC			
$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)	λ_1	$e(\sigma)$ (dB)	$e(\mathbf{a})$ (dB)	$e(\mathbf{l})$ (dB)
-55,94	-28,90	-18,81	0	-56,16	-28,90	-18,81
			0,1	-56,15	-28,90	-18,81
			1	-56,11	-28,98	-18,81
			5	-55,90	-29,31	-18,82
			10	-55,64	-29,73	-18,83
			25	-54,90	-30,94	-18,84
			50	-53,69	-32,26	-18,86
			70	-52,75	-31,92	-18,87
			75	-52,52	-31,63	-18,87
			80	-52,30	-31,28	-18,87
			100	-50,30	-27,15	-18,85

Observation tardive Nous venons d'observer que l'on est capable d'obtenir des valeurs positives de λ_1 , en utilisant la méthode GCV, lorsque l'on multiplie le signal $\bar{\mathbf{y}}$ par un facteur multiplicatif (essayé pour p.ex. un facteur 1000). Il existe donc possiblement un effet d'échelle qui provoque des problèmes numériques en calculant l'expression GCV sous sa forme pratique introduite en Annexe (dans la Sous-section E.1.5).

Aussi, en fin de stage, il reste donc à régler l'estimation de l'hyperparamètre dont l'effet semble en effet intéressant.

3. Expression de la vraisemblance

Comme dans le cas de la méthode R₁QPC, nous maximisons la vraisemblance $M_{N,R_{12}}$ d'avoir un modèle avec les paramètres $\Xi_{R_{12}}$ sous l'hypothèse d'observer le signal y :

$$\tilde{\Xi}_{R_{12}} = \arg \max_{\Xi_{R_{12}}} M_{N,R_{12}}, \text{ avec } M_{N,R_{12}} \triangleq p(\Xi_{R_{12}} | y, \mathcal{H}').$$

Nous utilisons la règle de Bayes, et introduisons une distribution a priori des paramètres \mathbf{a} égale à celle dans l'Équation (4.2), et une distribution a priori des paramètres \mathbf{l}

$$p(\Xi_{\mathbf{l},R_{12}}) = (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_{\mathbf{l}})^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{l}'^T \mathbf{R}_{\mathbf{l}}^{-1} \mathbf{l}'}, \quad (4.9)$$

avec la matrice de corrélation $\mathbf{R}_{\mathbf{l}}$ qui contient l'information sur le lisseur qui correspond à

$$\mathbf{R}_{\mathbf{l}}^{-1} = \frac{\lambda_2}{\sigma^2} \mathbf{\Lambda}'_k,$$

analogue au choix de la matrice $\mathbf{R}_{\mathbf{a}}$ dans l'Équation (4.3), et $\mathbf{l}' \in \mathbb{R}^{2K^2+K-1}$ qui égale $\mathbf{l} \in \mathbb{R}^{2K^2+K}$ sauf le premier élément l_{11} . En choisissant d'utiliser la matrice $\mathbf{\Lambda}'_k$ au lieu de $\mathbf{\Lambda}_k$ (et par conséquent de ne pas régulariser l'élément l_{11}), nous évitons que $\det \mathbf{R}_{\mathbf{l}}$ ne soit nul et avec lui la probabilité $M_{N,R_{12}}$.

Définissons $\Lambda'_k \triangleq \sqrt{\det \mathbf{\Lambda}'_k}$. Nous obtenons l'expression suivante pour la vraisemblance :

$$P'_N \cdot p(\Xi_{R_{12}}) = (2\pi)^{-\frac{3N}{2}} \frac{\lambda_1^K \lambda_2^{K^2+\frac{K}{2}-\frac{1}{2}}}{\sigma^{N+2K^2+3K-1}} [(2K)!]^k \Lambda'_k e^{-\frac{1}{2\sigma^2} (Q'_T + \lambda_1 \mathbf{a}'^T \mathbf{\Delta}_k \mathbf{a} + \lambda_2 \mathbf{l}'^T \mathbf{\Lambda}'_k \mathbf{l}')}$$

Nous définissons

$$Q_{T,R_{12}} \triangleq Q'_T + \lambda_1 \mathbf{a}'^T \mathbf{\Delta}_k \mathbf{a} + \lambda_2 \mathbf{l}'^T \mathbf{\Lambda}'_k \mathbf{l}'.$$

4. Maximisation de la vraisemblance

On peut prouver que les paramètres $\tilde{\Xi}_{R_{12}}$ sont la solution du système suivant :

$$\left\{ \begin{array}{l} \sigma = \sqrt{\frac{Q_{T,R_{12}}}{N + 2K^2 + 3K - 1}} \quad (4.10a) \\ R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k - \lambda_1 \cdot p^{2k} a_p + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \quad \forall p \in \llbracket 1, 2K \rrbracket \quad (4.10b) \\ S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j} \\ \quad - \lambda_2 \left(\sum_{k_1+k_2=k} (s-1)^{2k_1} (r-1)^{2k_2} \right) l_{r,s}, \quad \forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket. \quad (4.10c) \end{array} \right.$$

Écriture matricielle Les Équations (4.10) peuvent être écrites sous la forme matricielle suivante :

$$\left(\left[\begin{array}{c|c} \mathbf{G}_{R,R_{12}} & \mathbf{G}_{S,R_{12}} \\ \hline \mathbf{G}_{S,R_{12}}^\top & \mathbf{G}_{T,R_{12}} \end{array} \right] - \lambda_1 \underbrace{\left[\begin{array}{c|c} \Delta_k & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} \end{array} \right]}_{\triangleq \mathbf{D}_1} - \lambda_2 \underbrace{\left[\begin{array}{c|c} \mathbf{O} & \mathbf{O} \\ \hline \mathbf{O} & \Lambda_k \end{array} \right]}_{\triangleq \mathbf{D}_2} \right) \cdot \begin{bmatrix} \mathbf{a} \\ \mathbf{l} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{R,R_{12}} \\ \mathbf{H}_{S,R_{12}} \end{bmatrix},$$

ou encore

$$(\mathbf{G}_{R_{12}} - \lambda_1 \mathbf{D}_1 - \lambda_2 \mathbf{D}_2) \Xi_{\mathbf{a}, \mathbf{l}, R_{12}} = \mathbf{H}_{R_{12}},$$

abrégée comme

$$\mathbf{G}_{\lambda, R_{12}} \Xi_{\mathbf{a}, \mathbf{l}, R_{12}} = \mathbf{H}_{R_{12}}. \quad (4.11)$$

L'Équation (4.11) est linéaire en les paramètres.

Nous concluons que $\tilde{\Xi}_{R_{12}}$ peut être calculé des Équations (4.11) et (4.10a).

Implémentation pratique Comme dans le cas des méthodes QPC et R₁QPC, on utilise le vecteur $\bar{\mathbf{y}}$ et les matrices $\bar{\mathbf{Y}}_1$ et $\bar{\mathbf{Y}}_2$ (définis dans la Sous-section 3.1.2) pour l'implémentation pratique.

5. Estimation des hyperparamètres λ_1 et λ_2

Les valeurs des hyperparamètres λ_1 et λ_2 sont obtenues par la GCV :

$$(\tilde{\lambda}_1, \tilde{\lambda}_2) = \arg \min \text{GCV}'(\lambda_1, \lambda_2), \text{ avec } \text{GCV}(\lambda_1, \lambda_2) = \frac{\|(\mathbf{I} - \mathbf{M}')\bar{\mathbf{y}}\|_2}{\text{Tr}(\mathbf{I} - \mathbf{M}')}$$

et ici

$$\mathbf{M}' \triangleq \left[\begin{array}{c|c} \bar{\mathbf{Y}}_1 & \bar{\mathbf{Y}}_2 \end{array} \right] \cdot (\mathbf{G}_{R_{12}} - \lambda_1 \mathbf{D}_1 - \lambda_2 \mathbf{D}_2)^{-1} \cdot \begin{bmatrix} \bar{\mathbf{Y}}_1^\top \\ \bar{\mathbf{Y}}_2^\top \end{bmatrix}.$$

Implémentation pratique On observe que le calcul en PYTHON de \mathbf{M}' mène à des problèmes de mémoire à cause de la taille des matrices. Une reformulation de la GCV est proposée en annexe, qui résout ce problème.

4.3.2 Évaluation

Pour l'évaluation de la méthode R₁₂QPC, nous procédons comme pour l'évaluation de la méthode R₁QPC. Nous observons que nous ne pouvons calculer le minimum de la $\text{GCV}(\lambda_1, \lambda_2)$ à cause du fait que cette expression semble ne pas être convexe.

Nous concluons que la méthode R₁₂QPC, développée et implémentée comme discutée ici, ne fonctionne pas. À l'heure actuelle, nous n'avons pu trouver la raison de ce fait.

Chapitre 5

Test sur des corpus sonores cohérents

Après avoir établi les méthodes RQPC, nous pourrions les appliquer en identifiant une base de données de sons. Faute de temps, nous n'avons pu le faire à cette date. Néanmoins, nous mentionnons quelques remarques.

Nous procédons en deux étapes : l'identification d'une base de données de sons (Section 5.1) et l'utilisation de cette identification dans un algorithme de synthèse (Section 5.2).

5.1 Identification d'un corpus sonore cohérent

Nous considérons une base de données de sons instrumentaux [27]. Pour un instrument donné, comme par exemple une trompette ou un trombone, nous disposons d'un ensemble de sons de quelques secondes où l'on joue à chaque fois une note donnée à une intensité donnée (*piano*, *forte*, etc.).

Notre but est d'estimer un modèle non linéaire commun à tous les sons émis par un instrument, ou à tous les sons émis par une position de l'instrument (par exemple par position de la coulisse du trombone ou par configuration des pistons de la trompette. Voici quelques remarques.

Dimension du modèle En l'absence d'une manière d'estimation de la dimension du modèle, c'est-à-dire du nombre de résonances K , on pourrait s'inspirer des résultats de la littérature pour choisir la dimension du modèle.

Amplitude réelle des sons enregistrés Les amplitudes des sons dans la base de données [27] sont normalisées et nous n'avons pu trouver des informations sur la procédure suivie. Il faudrait choisir comment réétalonner les amplitudes selon les intensités.

Identification des vecteurs \mathbf{a} et \mathbf{l} Nous voulons estimer un modèle commun à plusieurs signaux y_1, \dots, y_P par la méthode RQPC. Pour chaque signal y_p , il est possible de construire les matrices $\mathbf{G}'_{\lambda,(p)}$ et $\mathbf{H}'_{(p)}$ de la méthode RQPC. Ensuite, ces matrices sont concaténées verticalement pour aboutir à l'expression

$$\tilde{\mathbf{G}}' \tilde{\mathbf{\Xi}}_{\mathbf{a},\mathbf{l}} = \tilde{\mathbf{H}}',$$

avec

$$\bar{\mathbf{G}}' \triangleq \begin{bmatrix} \mathbf{G}'_{\lambda,(1)} \\ \vdots \\ \mathbf{G}'_{\lambda,(p)} \\ \vdots \\ \mathbf{G}'_{\lambda,(P)} \end{bmatrix}, \quad \bar{\mathbf{H}}' \triangleq \begin{bmatrix} \mathbf{H}'_{(1)} \\ \vdots \\ \mathbf{H}'_{(p)} \\ \vdots \\ \mathbf{H}'_{(P)} \end{bmatrix}$$

et $\bar{\mathbf{E}}_{\mathbf{a},\mathbf{l}}$ les vecteurs \mathbf{a} et \mathbf{l} du modèle commun. Ceci représente un système d'équations linéaires surdéterminé, que l'on peut résoudre en utilisant la pseudo-inverse :

$$\bar{\mathbf{E}}_{\mathbf{a},\mathbf{l}} = (\bar{\mathbf{G}}'^{\top} \bar{\mathbf{G}}')^{-1} \bar{\mathbf{G}}'^{\top} \bar{\mathbf{H}}'.$$

Identification des entrées À partir du vecteur $\bar{\mathbf{E}}_{\mathbf{a},\mathbf{l}}$ du modèle commun, la méthode RQPC permet de calculer l'écart type de l'entrée gaussienne qui est aussi commune. Or, nous voulons différencier plusieurs sons générés par le modèle commun par leur entrée. Non seulement le fait de jouer plus ou moins fort, mais aussi la sélection de partiel pour une position donnée, ou même la sélection de la position doivent être représentés par l'entrée. Une piste pourrait être de calculer une entrée non gaussienne issue de la différence entre le son généré par le modèle commun et le vrai son.

5.2 Synthèse sonore

On peut appliquer un certain signal d'entrée au modèle commun identifié pour obtenir ainsi un son synthétisé. Comme mentionné chez l'identification des entrées dans la Section précédente, il faut étudier comment construire une entrée en fonction du son résultant désiré. On pourrait entre autres se renseigner sur les signaux MLI (pour Modulation de Largeur d'Impulsion, en anglais *PWM* pour *Pulse Width Modulation*).

Chapitre 6

Conclusion et perspectives

6.1 Conclusion

La méthode d'identification QPC établie pour un système quadratique en l'état, bien que fonctionnant en théorie, nécessite de très nombreux échantillons pour donner une estimation correcte. Nous avons présenté deux méthodes régularisées RQPC qui incorporent une régularisation sur le lisseur des deux premiers noyaux de transfert dans le but d'améliorer l'estimation du système. Néanmoins, l'implémentation de ces deux méthodes semble ne pas fonctionner, donc plus de recherche est nécessaire.

6.2 Perspectives

À l'issue de notre recherche, nous proposons les éléments suivants pour continuer le développement d'une méthode d'identification robuste pour une application à la synthèse sonore.

- Il faut valider les méthodes RQPC par une implémentation fonctionnelle. Si nécessaire, il faut adapter ces méthodes.
- Une étape supplémentaire serait d'étendre les méthodes proposées au modèle \mathcal{M} ; il faudrait tout d'abord regarder l'ajout de termes non linéaires $M_{p,0}$ ($p > 2$) (puissances de l'état), avant de s'intéresser aux termes généraux $M_{p,q}$ (monômes mixtes de l'état et l'entrée), qui ont une probabilité plus compliquée à cause du facteur γ (Éq. (A.1)) qui devient variable dans le temps.
- Dans le but de faire de la synthèse sonore, on veut appliquer la méthode à une base de données de sons instrumentaux. Pour cela, il est nécessaire de pouvoir estimer conjointement l'ordre $2K$ du système. Il faudrait considérer des recherches sur le critère d'Akaike [28], où l'on rajoute par exemple un terme $+\mu \cdot 2K$ dans la fonction de coût de la méthode, avec un hyperparamètre μ dont la valeur doit être calculée.
- Il faudrait assouplir l'Hypothèse **H2** en permettant une variation de l'écart type σ de la source. Ceci permettrait de modéliser une source dont l'amplitude varie dans le temps.
- On peut faire une étude théorique de l'influence de bruit sur la méthode d'identification, par exemple du bruit de sortie :

$$y[n] = x[n] + p[n], p[n] \sim \mathcal{N}(0, \sigma_p^2).$$

Annexe A

Développement de la Méthode LPC

Cette annexe contient les calculs explicités correspondants à la Sous-section 2.1.2. Elle est basée sur [16].

Nous procédons en deux étapes : l'expression de la vraisemblance (Section A.1) et la maximisation de la vraisemblance (Section A.2).

A.1 Expression de la vraisemblance

Nous voulons maximiser la vraisemblance P_N d'observer le signal y sous les hypothèses \mathcal{H} et sous l'hypothèse du modèle Θ :

$$\tilde{\Theta} \triangleq \arg \max_{\Theta} P_N, \text{ avec } P_N \triangleq p(y[0], \dots, y[N] | \mathcal{H}, \Theta).$$

1. Probabilité conditionnelle En utilisant la règle de la probabilité conditionnelle, $p(a,b|c) = p(a|b,c)p(b|c) = p(b|a,c)p(a|c)$, et en partitionnant $(y[1], \dots, y[N])$ en $a = (y[1], \dots, y[N-1])$ et $b = y[N]$, nous pouvons écrire

$$P_N = p(y[1], \dots, y[N] | \Theta, \mathcal{H}) = \underbrace{p(y[N] | y[1], \dots, y[N-1], \Theta, \mathcal{H})}_{\triangleq \pi_N} \cdot \underbrace{p(y[1], \dots, y[N-1] | \Theta, \mathcal{H})}_{= P_{N-1}}.$$

On peut continuer à appliquer cette règle de façon récursive. Tenu compte de l'hypothèse **H1**, nous obtenons donc $P_N = \pi_N P_{N-1} = \pi_N \pi_{N-1} P_{N-2} = \dots = \pi_N \cdot \dots \cdot \pi_2 P_1$, avec $P_1 = \pi_1$, ou

$$P_N = \prod_{n=1}^N \pi_n.$$

2. Changement de variable Ensuite, nous devons expliciter π_n , avant de pouvoir maximiser la vraisemblance P_N . La règle de changement de variable permet de faire cela :

$$\pi_n(y[n] | \tilde{H}) = \frac{1}{|\gamma|} \cdot p_U(f^{-1}(y[n]) | \tilde{H})$$

avec

$$\gamma \triangleq f' \circ f^{-1}(y[n]) \quad (\text{A.1})$$

et $\tilde{H} = \{y[1], \dots, y[n-1], \Theta, \mathcal{H}\}$. Quant au facteur γ , on a

$$y[n] = f(u[n]) = u[n] + \sum_{k=1}^{2K} a_k y[n-k],$$

$$u[n] = f^{-1}(y[n]) = y[n] - \sum_{k=1}^{2K} a_k y[n-k],$$

donc, puisque $y[n]$ est linéaire en $u[n]$,

$$\gamma = f'(f^{-1}(y[n])) = f'(u[n]) = 1.$$

En utilisant l'Hypothèse **H2**, nous obtenons

$$\begin{aligned} \pi_n &= p(y[n]|y[1], \dots, y[n-1], \Theta, \mathcal{H}) = p_U(u[n]|y[1], \dots, y[n-1], \Theta, \mathcal{H}) \\ &= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left[y[n] - \sum_{k=1}^{2K} a_k y[n-k]\right]^2}{2\sigma^2}\right). \end{aligned} \quad (\text{A.2})$$

Finalement, on trouve l'expression suivante pour la vraisemblance P_N :

$$P_N = \prod_{n=1}^N \pi_n = \left[\frac{1}{\sqrt{2\pi}\sigma}\right]^N \exp\left[-\frac{Q_{\mathbb{T}}(\Theta)}{2\sigma^2}\right], \quad (\text{A.3})$$

avec

$$Q_{\mathbb{T}}(\Theta) = \sum_{n \in \mathbb{T}} \left[\left[y[n] - \sum_{k=1}^{2K} a_k y[n-k] \right]^2 \right].$$

A.2 Maximisation de la vraisemblance

Nous définissons l'anti-log-vraisemblance $L_{\mathbb{T}} \triangleq -\ln P_N$, donc, de l'Équation (A.3),

$$L_{\mathbb{T}} \triangleq -\ln P_N = \frac{N}{2} \ln(2\pi) + N \ln \sigma + \frac{Q_{\mathbb{T}}}{2\sigma^2}. \quad (\text{A.4})$$

Puisque la fonction $-\ln(\cdot)$ est strictement décroissante, maximiser la vraisemblance P_N selon Θ correspond à minimiser l'anti-log-vraisemblance $L_{\mathbb{T}}$ selon Θ .

La fonction $L_{\mathbb{T}}$ est régulière, donc son gradient s'annule à ses extrema. Or, on peut prouver que l'extremum obtenu en annulant les dérivées de l'anti-log-vraisemblance selon les paramètres,

$$\frac{\partial L_{\mathbb{T}}}{\partial \Theta} = \mathbf{o}_{2K+1},$$

correspond à un minimum.

Nous pouvons donc finalement retrouver les Équations (2.3) par les dérivations suivantes.

Dérivée par rapport à σ

$$\frac{\partial L_{\mathbb{T}}}{\partial \sigma} = \frac{N}{\sigma^3} \left(\sigma^2 - \frac{Q_{\mathbb{T}}}{N} \right) = 0,$$

donc

$$\sigma = \sqrt{\frac{Q_{\mathbb{T}}}{N}}.$$

Dérivée par rapport à \mathbf{a}

$$\frac{\partial L_{\mathbb{T}}}{\partial a_p} = \frac{1}{2\sigma^2} \frac{\partial Q_{\mathbb{T}}}{\partial a_p} = \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] \right) y[n-p] \right] = \frac{1}{\sigma^2} \left[R_{p,0} - \sum_{k=1}^{2K} R_{p,k} a_k \right] = 0$$

avec

$$R_{p,k} \triangleq \sum_{n \in \mathbb{T}} y[n-p] y[n-k],$$

donc

$$R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k, \forall p \in \llbracket 1, 2K \rrbracket.$$

Annexe B

Développement de la Méthode QPC

Cette annexe contient les calculs explicités correspondants à la Sous-section 3.1.2.

Nous procédons en deux étapes : l'expression de la vraisemblance (Section B.1) et la maximisation de la vraisemblance (Section B.2).

B.1 Expression de la vraisemblance

Nous voulons maximiser la vraisemblance P_N d'observer le signal y sous les hypothèses \mathcal{H}' et sous l'hypothèse du modèle Ξ :

$$\tilde{\Xi} \triangleq \underset{\Xi}{\arg \max} P'_N, \text{ avec } P'_N \triangleq p(y[0], \dots, y[N] | \mathcal{H}', \Xi).$$

1. Probabilité conditionnelle De façon analogue à la Section A.1, nous obtenons que

$$P'_N = \prod_{n=1}^N \pi'_n.$$

2. Changement de variable Nous procédons comme dans la Section A.1. On a que

$$y[n] = f(u[n]) = u[n] + \sum_{k=1}^{2K} a_k y[n-k] + \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j],$$

$$u[n] = f^{-1}(y[n]) = y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j],$$

donc, puisque $y[n]$ est linéaire en $u[n]$, $\gamma = 1$ (cf. l'Équation (A.1)).

En utilisant l'Hypothèse **H2**, nous obtenons

$$\begin{aligned} \pi'_n &= p(y[n]|y[1], \dots, y[n-1], \Xi, \mathcal{H}') = p_U(u[n]|y[1], \dots, y[n-1], \Xi, \mathcal{H}') \\ &= \frac{1}{\sqrt{2\pi}\sigma} \exp \left(- \frac{\left[y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right]^2}{2\sigma^2} \right). \end{aligned} \quad (\text{B.1})$$

Finalement, on trouve l'expression suivante pour la vraisemblance P'_N :

$$P'_N = \prod_{n=1}^N \pi'_n = \left[\frac{1}{\sqrt{2\pi}\sigma} \right]^N \exp \left[- \frac{Q'_\mathbb{T}(\Theta)}{2\sigma^2} \right], \quad (\text{B.2})$$

avec

$$Q'_\mathbb{T}(\Theta) = \sum_{n \in \mathbb{T}} \left[\left[y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right]^2 \right]. \quad (\text{B.3})$$

B.2 Maximisation de la vraisemblance

Nous définissons l'anti-log-vraisemblance $L'_\mathbb{T} \triangleq -\ln P'_N$, donc, de l'Équation (B.2),

$$L'_\mathbb{T} \triangleq -\ln P'_N = \frac{N}{2} \ln(2\pi) + N \ln \sigma + \frac{Q'_\mathbb{T}}{2\sigma^2}. \quad (\text{B.4})$$

Puisque la fonction $-\ln(\cdot)$ est strictement décroissante, maximiser la vraisemblance P'_N selon Ξ correspond à minimiser l'anti-log-vraisemblance $L'_\mathbb{T}$ selon Ξ .

La fonction $L'_\mathbb{T}$ est régulière, donc son gradient s'annule à ses extrema. Or, on peut prouver que l'extremum obtenu en annulant les dérivées de l'anti-log-vraisemblance selon les paramètres,

$$\frac{\partial L'_\mathbb{T}}{\partial \Xi} = \mathbf{o}_{2K^2+3K+1},$$

correspond à un minimum.

Nous pouvons donc finalement retrouver les Équations (3.1) par les dérivations suivantes.

Dérivée par rapport à σ

$$\frac{\partial L'_\mathbb{T}}{\partial \sigma} = \frac{N}{\sigma^3} \left(\sigma^2 - \frac{Q'_\mathbb{T}}{N} \right) = 0,$$

donc

$$\sigma = \sqrt{\frac{Q'_\mathbb{T}}{N}}.$$

Dérivée par rapport à a

$$\begin{aligned} \frac{\partial L_{\mathbb{T}}}{\partial a_p} &= \frac{1}{2\sigma^2} \frac{\partial Q'_{\mathbb{T}}}{\partial a_p} = \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-p] \right] \\ &= \frac{1}{\sigma^2} \left[R_{0,p} - \sum_{k=1}^{2K} R_{p,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j} \right] = 0, \end{aligned}$$

avec

$$S_{\alpha,\beta,\gamma} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta] y[n-\gamma],$$

donc

$$R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \forall p \in \llbracket 1, 2K \rrbracket.$$

Dérivée par rapport à l

$$\begin{aligned} \frac{\partial L_{\mathbb{T}}}{\partial l_{r,s}} &= \frac{1}{2\sigma^2} \frac{\partial Q'_{\mathbb{T}}}{\partial l_{r,s}} \\ &= \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-r] y[n-s] \right] \\ &= \frac{1}{\sigma^2} \left[S_{0,r,s} - \sum_{k=1}^{2K} S_{r,s,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j} \right] = 0, \end{aligned}$$

avec

$$T_{\alpha,\beta,\gamma,\delta} \triangleq \sum_{n \in \mathbb{T}} y[n-\alpha] y[n-\beta] y[n-\gamma] y[n-\delta],$$

donc

$$S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j}, \forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket.$$

Annexe C

Idées pour l'amélioration de la méthode QPC

Lorsque l'on ne dispose que d'un nombre limité de données pour estimer une quantité, il faut essayer d'en extraire le maximum d'information possible [18]. Dans cette annexe, nous présentons quelques remarques sur deux idées que l'on a considérées dans le but de rendre la méthode QPC plus robuste : l'analyse par intervalle (Section C.1) et l'information de Fisher avec la borne de Cramer-Rao (Section C.2).

C.1 Analyse par intervalle

Lorsque l'on estime des paramètres par des estimateurs utilisant un nombre limité de données, il est important de rajouter une indication de la qualité de l'estimation [18]. Il est par exemple possible qu'un estimateur biaisé fournisse une estimation plus correcte pour un certain nombre limité de données qu'un estimateur non biaisé, qui donnerait une estimation exacte pour un nombre très grand (voire infini) de données.

L'analyse par intervalle permet de calculer des régions de confiance non asymptotiques pour une estimation d'un vecteur de paramètres [29]. Deux méthodes existent : la méthode LSCR (de l'anglais *Leave-out Sign-dominant Correlated Regions*, [30, 31]) pour le cas de l'estimation de paramètres d'un système non linéaire et la méthode SPS (de l'anglais *Sign-Perturbed Sums*, [32, 33, 34]) pour le cas de l'estimation de paramètres d'un système linéaire.

Dans le but d'améliorer la méthode QPC, il nous intéresse d'étudier la variance des matrices \mathbf{G}' et \mathbf{H}' , et non pas des paramètres du modèle Ξ . A priori, nous ne savons pas si l'on peut utiliser les méthodes citées pour ce problème.

C.2 Information de Fisher et borne de Cramer-Rao

C.2.1 Introduction

Considérons le cas où nous estimons un vecteur de paramètres θ basé sur un signal d'observations y . La matrice d'information de Fisher $\mathcal{I}(\theta)$ [35] représente la quantité d'information sur les paramètres que le signal contient. Elle s'exprime comme

$$[\mathcal{I}(\boldsymbol{\theta})]_{i,j} \triangleq -\mathbb{E} \left[\frac{\partial^2}{\partial \theta_i \partial \theta_j} \ln p(y|\boldsymbol{\theta}) \right]. \quad (\text{C.1})$$

Cette matrice est utilisée dans la Borne de Cramer-Rao [36, 37], qui est une borne inférieure sur la variance d'un estimateur asymptotique Ξ :

$$\text{Cov}(\boldsymbol{\theta}) \geq C \cdot (\mathcal{I}(\boldsymbol{\theta}))^{-1}, \quad (\text{C.2})$$

où nous ne précisons pas l'expression C .

Il nous intéresse d'utiliser la borne (Équation (C.2)) pour évaluer si l'on l'atteint avec la méthode QPC. Si c'était le cas, on sait que l'on ne pourrait pas l'améliorer.

C.2.2 Calcul pour le modèle \mathcal{M}_1

Dans ce cas, $p(y|\boldsymbol{\theta}) = P_N$ et $\boldsymbol{\theta} = \boldsymbol{\Theta}$ dans les Équations (C.1)-(C.2).

Propriétés statistiques de la sortie On peut montrer que les deux propriétés suivantes sont vérifiées :

$$\begin{aligned} \mathbb{E}[y[n]|\boldsymbol{\Theta}] &= 0, \quad \forall n, \\ \mathbb{E}[y[n]y[n-k]|\boldsymbol{\Theta}] &= \sigma^2 \sum_{l=\max(0,k)}^n h_l h_{l-k}, \quad \forall n, \forall k, \end{aligned}$$

où h est la réponse impulsionnelle du système (qui peut se trouver à partir du vecteur \mathbf{a}). On a donc :

$$\mathbb{E}[R_{p,q}|\boldsymbol{\Theta}] = \sigma^2 \mathcal{R}_{p,q}^E \quad \text{avec} \quad \mathcal{R}_{p,q}^E \triangleq \sum_{n \in \mathbb{T}} \sum_{l=\max(p,q)}^n h_{l-p} h_{l-q}$$

De même :

$$\begin{aligned} \mathbb{E}[Q_{\mathbb{T}}|\boldsymbol{\Theta}] &= \sum_{n \in \mathbb{T}} \mathbb{E} \left[\left[y[n] - \sum_{k=1}^{2K} a_k y[n-k] \right]^2 \middle| \boldsymbol{\Theta} \right] \\ &= \sum_{n \in \mathbb{T}} \left(\mathbb{E}[y^2[n]|\boldsymbol{\Theta}] - \mathbb{E} \left[2y[n] \sum_{k=1}^{2K} a_k y[n-k] \middle| \boldsymbol{\Theta} \right] + \mathbb{E} \left[\sum_{k=1}^{2K} \sum_{l=1}^{2K} a_k a_l y[n-k] y[n-l] \middle| \boldsymbol{\Theta} \right] \right) \\ &= \sum_{n \in \mathbb{T}} \left(\sigma^2 - 2 \sum_{k=1}^{2K} a_k \mathbb{E}[y[n]y[n-k]|\boldsymbol{\Theta}] + \sum_{k=1}^{2K} \sum_{l=1}^{2K} a_k a_l \mathbb{E}[y[n-k]y[n-l]|\boldsymbol{\Theta}] \right) \\ &= \sigma^2 \left(N - 2 \sum_{k=1}^{2K} a_k \mathcal{R}_{0,k}^E + \sum_{k,l=1}^{2K} a_k a_l \mathcal{R}_{k,l}^E \right) \end{aligned}$$

Information de Fisher Partant des calculs dans l'Annexe A, on peut calculer que

$$\begin{aligned}
-\frac{\partial^2}{\partial \Theta^2} \ln P_N &= \begin{bmatrix} \frac{\partial^2 L_{\mathbb{T}}}{\partial \sigma^2} & \frac{\partial^2 L_{\mathbb{T}}}{\partial \sigma \partial a_p} \\ \frac{\partial^2 L_{\mathbb{T}}}{\partial a_p \partial \sigma} & \frac{\partial^2 L_{\mathbb{T}}}{\partial a_q \partial a_p} \end{bmatrix} \\
&= \begin{bmatrix} -\frac{N}{\sigma^2} + 3\frac{Q_{\mathbb{T}}}{\sigma^4} & -\frac{2}{\sigma^3} \left(R_{p,0} - \sum_{k=1}^{2K} R_{p,k} a_k \right) \\ -\frac{2}{\sigma^3} \left(R_{p,0} - \sum_{k=1}^{2K} R_{p,k} a_k \right) & -\frac{1}{\sigma^2} R_{p,q} \end{bmatrix},
\end{aligned}$$

donc :

$$\begin{aligned}
\mathcal{I}(\Theta) &= -\mathbb{E} \left(\frac{\partial^2}{\partial \Theta^2} \ln P_N \right) \\
&= \begin{bmatrix} 2\frac{N}{\sigma^2} - \frac{6}{\sigma^2} \sum_{k=1}^{2K} a_k \mathcal{R}_{0,k}^{\mathbb{E}} + \frac{3}{\sigma^2} \sum_{k,l=1}^{2K} a_k a_l \mathcal{R}_{k,l}^{\mathbb{E}} & -\frac{2}{\sigma} \left(\mathcal{R}_{p,0}^{\mathbb{E}} - \sum_{k=1}^{2K} \mathcal{R}_{p,k}^{\mathbb{E}} a_k \right) \\ -\frac{2}{\sigma} \left(\mathcal{R}_{p,0}^{\mathbb{E}} - \sum_{k=1}^{2K} \mathcal{R}_{p,k}^{\mathbb{E}} a_k \right) & -\mathcal{R}_{p,q}^{\mathbb{E}} \end{bmatrix}
\end{aligned}$$

Borne de Cramer-Rao On peut inverser $\mathcal{I}(\theta)$ en utilisant les propriétés de l'inversion matricielle par bloc. Mais au vu de la formule de la matrice d'information de Fisher, le calcul serait fastidieux, et le résultat probablement difficilement exploitable.

C.2.3 Calcul pour le modèle \mathcal{M}_2

Dans ce cas, $p(y|\theta) = P'_N$ et $\theta = \Xi$ dans les Équations (C.1)-(C.2).

Information de Fisher Partant des calculs dans l'Annexe B, on peut calculer que

$$\begin{aligned}
-\frac{\partial^2}{\partial \Xi^2} \ln P'_N &= \begin{bmatrix} \frac{\partial^2 L'_{\mathbb{T}}}{\partial \sigma^2} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial \sigma \partial a_p} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial \sigma \partial L_{p,q}} \\ \frac{\partial^2 L'_{\mathbb{T}}}{\partial a_p \partial \sigma} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial a_q \partial a_p} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial a_l \partial L_{p,q}} \\ \frac{\partial^2 L'_{\mathbb{T}}}{\partial L_{p,q} \partial \sigma} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial L_{p,q} \partial a_p} & \frac{\partial^2 L'_{\mathbb{T}}}{\partial L_{l,m} \partial L_{p,q}} \end{bmatrix} \\
&= \begin{bmatrix} -\frac{N}{\sigma^2} + 3\frac{Q'_{\mathbb{T}}}{\sigma^4} & \text{sym} & \text{sym} \\ -\frac{2}{\sigma^3} \left(R_{p,0} - \sum_{k=1}^{2K} R_{p,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} L_{i,j} \right) & -\frac{1}{\sigma^2} R_{p,q} & \text{sym} \\ -\frac{2}{\sigma^3} \left(S_{p,q,0} - \sum_{k=1}^{2K} S_{p,q,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{p,q,i,j} L_{i,j} \right) & -\frac{1}{\sigma^2} S_{p,q,l} & -\frac{1}{\sigma^2} T_{p,q,l,m} \end{bmatrix},
\end{aligned}$$

où « sym » indique un élément égal à son élément symétrique dans la matrice.

Borne de Cramer-Rao Comme dans le cas linéaire, nous arrêtons le calcul à ce point parce qu'il est fastidieux et nous semble difficilement exploitable.

Annexe D

Région de convergence

De façon analogue à un résultat dans [22] pour le cas du temps continu, on peut établir une expression pour une borne sur l'entrée du système considéré qui assure que l'on reste dans la région de convergence :

$$\|u\|_\infty < \frac{1}{\epsilon_p \alpha_p \|\mathbf{h}_1\|_{\mathcal{V}_p^1}},$$

avec

$$\begin{aligned}\epsilon_p &\triangleq \|\mathbf{B}\|_p \|\mathbf{L}\|_p \\ \alpha_p &\triangleq \sum_{m=0}^{+\infty} \|\mathbf{A}^m\|_p \\ \|\mathbf{h}_1\|_{\mathcal{V}_p^1} &\triangleq \sum_{k=0}^{+\infty} \|\mathbf{h}_1[k]\|_p \\ \mathbf{h}_1[k] &\triangleq \mathbf{A}^k \mathbf{B}.\end{aligned}$$

Cette borne est possiblement conservatrice. Parce que l'on n'étudie pas la région de convergence pendant ce stage, nous n'avons pas pris en compte cette borne.

Annexe E

Développement de la Méthode RQPC

Cette annexe contient les calculs explicités correspondants au Chapitre 4.

E.1 Développement de la Méthode R₁QPC

E.1.1 Calcul du noyau de transfert de Volterra d'ordre 1

La méthode R₁QPC est la méthode QPC avec une régularisation sur le noyau de transfert de Volterra d'ordre 1, $H_1(z)$, que nous calculons ici pour le modèle \mathcal{M}_2 . De l'Équation (2.10) suit que

$$H_1(z) = \mathbf{C}\mathbf{W}(z)\mathbf{B}, \text{ avec } \mathbf{W}(z) = (z\mathbf{I} - \mathbf{A})^{-1}.$$

La multiplication par les matrices \mathbf{C} et \mathbf{B} correspond à une sélection de l'élément $(2K, 2K)$ de la matrice $\mathbf{W}(z)$ (cf. forme des matrices dans l'Éq. (2.5)). Nous pouvons écrire cette matrice comme une matrice par blocs :

$$\mathbf{W} = (z\mathbf{I} - \mathbf{A})^{-1} = \left[\begin{array}{ccc|c} z & -1 & & \\ & z & -1 & \\ & & \ddots & \\ & & & z & -1 \\ \hline -a_{2K} & -a_{2K-1} & \dots & -a_2 & (z - a_1) \end{array} \right]^{-1} \triangleq \left[\begin{array}{c|c} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{array} \right]^{-1}.$$

Cette inversion par bloc s'exprime comme

$$\left[\begin{array}{c|c} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{array} \right]^{-1} = \left[\begin{array}{c|c} \mathbf{E}_{11}^{-1} + \mathbf{E}_{11}^{-1}\mathbf{E}_{12}\mathbf{F}\mathbf{E}_{21}\mathbf{E}_{11}^{-1} & -\mathbf{E}_{11}^{-1}\mathbf{E}_{12}\mathbf{F} \\ \hline -\mathbf{F}\mathbf{E}_{21}\mathbf{E}_{11}^{-1} & \mathbf{F} \end{array} \right],$$

où $\mathbf{F} \triangleq (\mathbf{E}_{22} - \mathbf{E}_{21}\mathbf{E}_{11}^{-1}\mathbf{E}_{12})^{-1}$. Nous calculons

$$\mathbf{E}_{11}^{-1} = \begin{bmatrix} z^{-1} & z^{-2} & \dots & z^{-2K+1} \\ & \ddots & \ddots & \vdots \\ & & \ddots & z^{-2} \\ & & & z^{-1} \end{bmatrix},$$

$$\begin{aligned}
 \mathbf{F} &= \begin{bmatrix} (z - a_1) - [-a_{2K} & -a_{2K-1} & \dots & -a_2] \cdot \begin{bmatrix} z^{-1} & z^{-2} & \dots & z^{-2K+1} \\ & \ddots & \ddots & \vdots \\ & & \ddots & z^{-2} \\ & & & z^{-1} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -1 \end{bmatrix} \end{bmatrix}^{-1} \\
 &= \frac{1}{(z - a_1) - \sum_{k=2}^{2K} a_k z^{-k+1}} = \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)},
 \end{aligned}$$

donc le noyau de transfert Volterra d'ordre 1

$$H_1(z) = \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)}. \quad (\text{E.1})$$

E.1.2 A priori de lisseur

Définissons

$$P(z) \triangleq \sum_{k=1}^{2K} a_k z^{-k}$$

dans l'Équation (E.1). Comme dans [20, 19], nous cherchons à régulariser le spectre suivant :

$$P(f) \triangleq P(e^{-j2\pi f}) = \sum_{p=1}^{2K} a_p e^{j2\pi f p} \triangleq \mathbf{e}^\top \mathbf{a}.$$

Pour cela, on définit le terme de « facteur de lissage linéaire d'ordre k » $R_k^{(1)}$ comme suit :

$$\begin{aligned}
 R_k^{(1)} &\triangleq \int_0^1 \left| \frac{d^k}{df^k} P(f) \right|^2 df \\
 &= \int_0^1 \left| \frac{d^k}{df^k} \mathbf{e}^\top \mathbf{a} \right|^2 df \\
 &= \int_0^1 \left| (j2\pi)^k \mathbf{e}^\top \bar{\Delta}_k \mathbf{a} \right|^2 df
 \end{aligned}$$

avec

$$\bar{\Delta}_k \triangleq \begin{bmatrix} 1^k & 0 & 0 & \dots & 0 \\ 0 & 2^k & 0 & \dots & 0 \\ 0 & 0 & 3^k & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & (2K)^k \end{bmatrix}.$$

Donc :

$$\begin{aligned}
R_k^{(1)} &= (j2\pi)^{2k} \int_0^1 (\mathbf{e}^\top \bar{\Delta}_k \mathbf{a})^H (\mathbf{e}^\top \bar{\Delta}_k \mathbf{a}) \, df \\
&= (-4\pi^2)^k \int_0^1 \mathbf{a}^\top \bar{\Delta}_k \mathbf{e}^* \mathbf{e}^\top \bar{\Delta}_k \mathbf{a} \, df \\
&= (-4\pi^2)^k \mathbf{a}^\top \bar{\Delta}_k \left(\int_0^1 \mathbf{e}^* \mathbf{e}^\top \, df \right) \bar{\Delta}_k \mathbf{a} \\
&= (-4\pi^2)^k \mathbf{a}^\top \bar{\Delta}_k \mathbf{I} \bar{\Delta}_k \mathbf{a} \\
&= (-4\pi^2)^k \mathbf{a}^\top \bar{\Delta}_k^2 \mathbf{a}
\end{aligned}$$

en utilisant le fait que

$$\int_0^1 e^{j2\pi f k} \, df = \begin{cases} 1 & \text{si } k = 0 \\ 0 & \text{si } k \in \mathbb{N}^*. \end{cases}$$

Ou, avec la définition $\Delta_k \triangleq \bar{\Delta}_k^2$ et donc

$$\Delta_k \triangleq \begin{bmatrix} 1^{2k} & 0 & 0 & \dots & 0 \\ 0 & 2^{2k} & 0 & \dots & 0 \\ 0 & 0 & 3^{2k} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & (2K)^{2k} \end{bmatrix}, \quad (\text{E.2})$$

$$R_k^{(1)} = (-4\pi^2)^k \mathbf{a}^\top \Delta_k \mathbf{a}.$$

Dérivée par rapport à \mathbf{a} Cette dérivée sera utilisée plus bas. On a que

$$\frac{dR_k^{(1)}}{d\mathbf{a}} = (-4\pi^2)^k 2\Delta_k \mathbf{a}$$

ou de manière équivalente :

$$\frac{\partial R_k^{(1)}}{\partial a_p} = (-4\pi^2)^k 2p^{2k} a_p. \quad (\text{E.3})$$

E.1.3 Expression de la vraisemblance

Afin de pouvoir inclure une distribution a priori sur les paramètres dans la fonction de coût, nous ne voulons pas maximiser la vraisemblance P'_N d'observer le signal y sous l'hypothèse du modèle Ξ (ce qui était le cas dans la méthode QPC, cf. Annexe B), mais nous voulons maximiser la vraisemblance M_{N,R_1} d'avoir un modèle avec les paramètres Ξ_{R_1} sous l'hypothèse d'observer le signal y :

$$\tilde{\Xi}_{R_1} \triangleq \arg \max_{\Xi_{R_1}} M_{N,R_1}, \text{ avec } M_{N,R_1} \triangleq p(\Xi_{R_1} | y, \mathcal{H}').$$

Dès maintenant, nous n'écrivons plus explicitement l'hypothèse \mathcal{H}' afin d'alléger la notation.

1. Règle de Bayes Le lien entre la probabilité M_{N,R_1} et P'_N est donné par la règle de Bayes :

$$M_{N,R_1} = \frac{P'_N \cdot p(\Xi_{R_1})}{p(y)}.$$

Puisque la probabilité $p(y)$ ne dépend pas des paramètres Ξ_{R_1} , c'est une constante multiplicative que nous pouvons enlever de la fonction de coût. Nous obtenons donc

$$\tilde{\Xi}_{R_1} = \arg \max_{\Xi_{R_1}} [P'_N \cdot p(\Xi_{R_1})].$$

2. Distribution a priori des paramètres La méthode R_1 QPC inclut une distribution a priori sur les coefficients linéaires \mathbf{a} dans la fonction de coût :

$$p(\Xi_{R_1}) = (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_a)^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{a}^\top \mathbf{R}_a^{-1} \mathbf{a}} \quad (\text{E.4})$$

avec la matrice de corrélation \mathbf{R}_a qui contient l'information sur le lisseur et qui correspond à

$$\mathbf{R}_a^{-1} = \frac{\lambda_1}{\sigma^2} \mathbf{\Delta}_k, \quad (\text{E.5})$$

comme dans [20].

3. Expression de la vraisemblance En utilisant les expressions pour P'_N (Éq. (B.2)) et pour $p(\Xi_{R_1})$ (Éq. (E.4)), nous pouvons écrire

$$P'_N \cdot p(\Xi_{R_1}) = \left[\frac{1}{\sqrt{2\pi}\sigma} \right]^N \exp \left[-\frac{Q'_\mathbb{T}(\Theta)}{2\sigma^2} \right] \cdot (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_a)^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{a}^\top \mathbf{R}_a^{-1} \mathbf{a}}.$$

La matrice $\mathbf{\Delta}_k$ (Eq. (E.2)) est diagonale. On calcule que

$$\begin{aligned} (\det \mathbf{R}_a)^{-\frac{1}{2}} &= \left(\frac{1}{\det \mathbf{R}_a^{-1}} \right)^{-\frac{1}{2}} = \sqrt{\det \left(\frac{\lambda_1}{\sigma^2} \mathbf{\Delta}_k \right)} = \sqrt{\frac{\lambda_1^{2K}}{\sigma^{4K}} \det \mathbf{\Delta}_k} = \frac{\lambda_1^K}{\sigma^{2K}} \sqrt{[(2K)!]^{2k}} \\ &= \frac{\lambda_1^K}{\sigma^{2K}} [(2K)!]^k, \end{aligned}$$

donc on a

$$P'_N \cdot p(\Xi_{R_1}) = (2\pi)^{-N} \frac{\lambda_1^K}{\sigma^{2K+N}} [(2K)!]^k e^{-\frac{1}{2\sigma^2} (Q'_\mathbb{T} + \lambda_1 \mathbf{a}^\top \mathbf{\Delta}_k \mathbf{a})}.$$

Nous définissons

$$Q_{\mathbb{T},R_1} \triangleq Q'_\mathbb{T} + \lambda_1 \mathbf{a}^\top \mathbf{\Delta}_k \mathbf{a}.$$

E.1.4 Maximisation de la vraisemblance

Nous définissons l'anti-log-vraisemblance

$$L_{\mathbb{T},R_1} \triangleq -\ln (P'_N \cdot p(\Xi_{R_1})) = N \ln 2\pi - K \ln \lambda_1 + (2K + N) \ln \sigma - k \ln [(2K)!] + \frac{1}{2\sigma^2} Q_{\mathbb{T},R_1}.$$

Comme dans le cas de la QPC, nous obtenons les Équations (4.4) pour les paramètres Ξ_{R_1} en annulant les dérivées de l'anti-log-vraisemblance selon les paramètres,

$$\frac{\partial L_{\mathbb{T},R_1}}{\partial \Xi_{R_1}} = \mathbf{o}_{2K^2+3K+1}.$$

Dérivée par rapport à σ

$$\frac{\partial L_{\mathbb{T}, R_1}}{\partial \sigma} = \frac{N + 2K}{\sigma^3} \left(\sigma^2 - \frac{Q_{\mathbb{T}, R_1}}{N + 2K} \right) = 0,$$

donc

$$\sigma = \sqrt{\frac{Q_{\mathbb{T}, R_1}}{N + 2K}}.$$

Dérivée par rapport à a

$$\frac{\partial L_{\mathbb{T}, R_1}}{\partial a_p} = \frac{1}{2\sigma^2} \frac{\partial Q_{\mathbb{T}, R_1}}{\partial a_p},$$

en utilisant l'Équation (E.3),

$$\begin{aligned} & \frac{\partial L_{\mathbb{T}, R_1}}{\partial a_p} \\ &= \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-p] \right] + \frac{1}{\sigma^2} \lambda_1 \cdot p^{2k} a_p \\ &= \frac{1}{\sigma^2} \left[R_{0,p} - \sum_{k=1}^{2K} R_{p,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j} + \lambda_1 \cdot p^{2k} a_p \right] = 0, \end{aligned}$$

donc

$$R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k - \lambda_1 \cdot p^{2k} a_p + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \forall p \in \llbracket 1, 2K \rrbracket.$$

Le terme $-\lambda_1 \cdot p^{2k} a_p$ est supplémentaire par rapport à la méthode QPC.

Dérivée par rapport à l

$$\begin{aligned} & \frac{\partial L_{\mathbb{T}, R_1}}{\partial l_{r,s}} = \frac{1}{2\sigma^2} \frac{\partial Q_{\mathbb{T}, R_1}}{\partial l_{r,s}} \\ &= \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-r] y[n-s] \right] \\ &= \frac{1}{\sigma^2} \left[S_{0,r,s} - \sum_{k=1}^{2K} S_{r,s,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j} \right] = 0, \end{aligned}$$

donc

$$S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j}, \forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket.$$

Cette équation est identique au cas de la QPC.

E.1.5 Estimation de l'hyperparamètre λ_1

Il nous reste à estimer la valeur $\tilde{\lambda}_1$ de l'hyperparamètre λ_1 , présent dans les formules de la méthode R_1 QPC. Pour le faire, nous adaptons la méthode GCV, utilisée dans [20, 26] dans le cas d'un système linéaire, à notre cas. L'hyperparamètre est trouvé en minimisant une expression dite GCV :

$$\tilde{\lambda}_1 = \arg \min \text{GCV}(\lambda_1).$$

Cette expression est définie comme

$$\text{GCV}(\lambda_1) = \frac{\|(\mathbf{I} - \mathbf{M})\bar{\mathbf{y}}\|_2}{\text{Tr}(\mathbf{I} - \mathbf{M})}, \quad (\text{E.6})$$

avec (cf. notation introduite en Sous-section 3.1.2)

$$\mathbf{M} \triangleq \begin{bmatrix} \bar{\mathbf{Y}}_1 & | & \bar{\mathbf{Y}}_2 \end{bmatrix} \cdot (\mathbf{G}_{R_1} - \lambda_1 \mathbf{D}_1)^{-1} \cdot \begin{bmatrix} \bar{\mathbf{Y}}_1^\top \\ \bar{\mathbf{Y}}_2^\top \end{bmatrix}.$$

On note que le numérateur de la GCV exprime la différence entre le signal de sortie y mesuré et la sortie prédite par le modèle obtenu par la méthode R_1 QPC :

$$\mathbf{M}\bar{\mathbf{y}} = \begin{bmatrix} \bar{\mathbf{Y}}_1 & | & \bar{\mathbf{Y}}_2 \end{bmatrix} \cdot (\mathbf{G}_{R_1} - \lambda \mathbf{D}_1)^{-1} \cdot \mathbf{H}_{R_1} = \begin{bmatrix} \bar{\mathbf{Y}}_1 & | & \bar{\mathbf{Y}}_2 \end{bmatrix} \cdot \Xi_{a,l,R_1},$$

ce qui est l'estimation du vecteur $\bar{\mathbf{y}}$.

Implémentation pratique Nous voulons calculer l'expression GCV (Équation (E.6)) en PYTHON. Réécrivons \mathbf{M} comme

$$\mathbf{M} = \bar{\mathbf{Y}} \mathbf{G}_{\lambda,R_1}^{-1} \bar{\mathbf{Y}}^\top.$$

On observe que le calcul de \mathbf{M} en exécutant les multiplications matricielles comme $(\bar{\mathbf{Y}} \mathbf{G}_{\lambda,R_1}^{-1}) \bar{\mathbf{Y}}^\top$ ou $\bar{\mathbf{Y}} (\mathbf{G}_{\lambda,R_1}^{-1} \bar{\mathbf{Y}}^\top)$ (en calculant d'abord ce qui est entre parenthèses) mène à des problèmes de mémoire à cause de la taille des matrices. Nous établissons ici une formulation alternative qui résout ce problème.

On a que

$$\begin{aligned} \|(\mathbf{I} - \mathbf{M})\bar{\mathbf{y}}\|_2^2 &= [(\mathbf{I} - \mathbf{M})\bar{\mathbf{y}}]^\top [(\mathbf{I} - \mathbf{M})\bar{\mathbf{y}}] \\ &= \bar{\mathbf{y}}^\top (\mathbf{I} - \mathbf{M})^\top (\mathbf{I} - \mathbf{M}) \bar{\mathbf{y}} \\ &= \bar{\mathbf{y}}^\top (\mathbf{I} - \mathbf{M}^\top) (\mathbf{I} - \mathbf{M}) \bar{\mathbf{y}} \\ &= \bar{\mathbf{y}}^\top (\mathbf{I} - 2\mathbf{M} + \mathbf{M}^2) \bar{\mathbf{y}} \text{ parce que } \mathbf{M} \text{ est symétrique} \\ &= \bar{\mathbf{y}}^\top \bar{\mathbf{y}} - 2(\bar{\mathbf{y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{y}}) + (\bar{\mathbf{y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{y}}) \end{aligned}$$

(en calculant d'abord ce qui est entre parenthèses) et

$$\text{Tr}(\mathbf{I} - \mathbf{M}) = \text{Tr}(\mathbf{I} - \bar{\mathbf{Y}} \mathbf{G}_{\lambda,R_1}^{-1} \bar{\mathbf{Y}}^\top) = \text{Tr}(\mathbf{I}) - \text{Tr}(\bar{\mathbf{Y}} \mathbf{G}_{\lambda,R_1}^{-1} \bar{\mathbf{Y}}^\top) = (N - 1) - \text{Tr}(\mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}}))$$

parce que la trace d'une matrice est invariante sous une permutation cyclique.

Nous obtenons finalement l'expression

$$\text{GCV}(\lambda_1) = \frac{\sqrt{\bar{\mathbf{y}}^\top \bar{\mathbf{y}} - 2(\bar{\mathbf{y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{y}}) + (\bar{\mathbf{y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}}) \mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{y}})}}{(N - 1) - \text{Tr}(\mathbf{G}_{\lambda,R_1}^{-1} (\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}}))}.$$

E.2 Développement de la Méthode R_{12} QPC

E.2.1 Calcul du noyau de transfert de Volterra d'ordre 2

En outre de la régularisation sur le noyau de transfert de Volterra d'ordre 1 dans la méthode R_1 QPC, la méthode R_{12} QPC inclut aussi une régularisation sur le noyau de transfert de Volterra d'ordre 2. Nous calculons ce noyau $H_2(z_1, z_2)$ pour le modèle \mathcal{M}_2 ici. De l'Équation (2.11) suit que

$$\begin{aligned} H_2(z_1, z_2) &\triangleq \mathbf{C}\mathbf{W}(z_1 + z_2)\mathbf{B}M_{2,0}(\mathbf{W}(z_1)\mathbf{B}, \mathbf{W}(z_2)\mathbf{B}) \\ &= [\mathbf{C}\mathbf{W}(z_1 + z_2)\mathbf{B}] \cdot [\mathbf{B}^\top \mathbf{W}(z_1)^\top] \cdot \mathbf{L} \cdot [\mathbf{W}(z_2)\mathbf{B}]. \end{aligned}$$

En utilisant des notations et résultats de Sous-section E.1.1, nous calculons les ingrédients suivants :

$$\begin{aligned} -\mathbf{E}_{11}^{-1}\mathbf{E}_{12}\mathbf{F} &= - \begin{bmatrix} z^{-1} & z^{-2} & \dots & z^{-2K+1} \\ & \ddots & \ddots & \vdots \\ & & \ddots & z^{-2} \\ & & & z^{-1} \end{bmatrix} \cdot \begin{bmatrix} \\ \\ \\ -1 \end{bmatrix} \cdot \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)} \\ &= \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)} \begin{bmatrix} z^{-2K+1} \\ \vdots \\ z^{-2} \\ z^{-1} \end{bmatrix}, \end{aligned}$$

donc

$$\begin{aligned} \mathbf{W}(z) &= \frac{1}{z \left(1 - \sum_{k=1}^{2K} a_k z^{-k} \right)} \begin{bmatrix} \text{pas relevant} & \left| \begin{array}{c} z^{-2K+1} \\ \vdots \\ z^{-1} \end{array} \right. \\ \text{pas relevant} & \left| \begin{array}{c} \phantom{z^{-2K+1}} \\ \\ 1 \end{array} \right. \end{bmatrix}; \\ \mathbf{C}\mathbf{W}(z_1 + z_2)\mathbf{B} &= \frac{1}{(z_1 + z_2) \left(1 - \sum_{k=1}^{2K} a_k (z_1 + z_2)^{-k} \right)}; \\ \mathbf{B}^\top \mathbf{W}(z_1)^\top &= \frac{1}{z_1 \left(1 - \sum_{k=1}^{2K} a_k z_1^{-k} \right)} \left[\begin{array}{ccc|c} z_1^{-2K+1} & \dots & z_1^{-1} & 1 \end{array} \right]; \\ \mathbf{W}(z_2)\mathbf{B} &= \frac{1}{z_2 \left(1 - \sum_{k=1}^{2K} a_k z_2^{-k} \right)} \left[\begin{array}{ccc|c} z_2^{-2K+1} & \dots & z_2^{-1} & 1 \end{array} \right]^\top. \end{aligned}$$

Nous obtenons finalement le noyau de transfert de Volterra d'ordre 2

$$H_2(z_1, z_2) = \frac{\sum_{p=1}^{2K} \sum_{q=p}^{2K} l_{pq} z_1^{1-q} z_2^{1-p}}{(z_1 + z_2) \left(1 - \sum_{k=1}^{2K} a_k (z_1 + z_2)^{-k} \right) z_1 \left(1 - \sum_{k=1}^{2K} a_k z_1^{-k} \right) z_2 \left(1 - \sum_{k=1}^{2K} a_k z_2^{-k} \right)}. \quad (\text{E.7})$$

Dérivée par rapport à \mathbf{l} Cette dérivée sera utilisée plus bas. On a que

$$\frac{dR_k^{(2)}}{d\mathbf{l}_{\alpha\beta}} = (4\pi^2)^k 2l_{\alpha\beta} \left(\sum_{k_1+k_2=k} (\beta-1)^{2k_1} (\alpha-1)^{2k_2} \right) \quad (\text{E.8})$$

ou de manière équivalente :

$$\frac{\partial R_k^{(2)}}{\partial \mathbf{l}} = (4\pi^2)^k 2\mathbf{\Lambda}_k \mathbf{l}.$$

Nous définissons également la matrice

$$\mathbf{\Lambda}'_k \in \mathbb{R}^{(2K^2+K-1) \times (2K^2+K-1)} \quad (\text{E.9})$$

qui égale la matrice $\mathbf{\Lambda}_k \in \mathbb{R}^{(2K^2+K) \times (2K^2+K)}$ sauf les première colonne et première ligne, qui sont nulles.

E.2.3 Expression de la vraisemblance

Comme dans le cas de la méthode R_1 QPC, nous maximons la vraisemblance $M_{N,R_{12}}$ d'avoir un modèle avec les paramètres $\Xi_{R_{12}}$ sous l'hypothèse d'observer le signal y :

$$\tilde{\Xi}_{R_{12}} = \arg \max_{\Xi_{R_{12}}} M_{N,R_{12}}, \text{ avec } M_{N,R_{12}} \triangleq p(\Xi_{R_{12}} | y, \mathcal{H}').$$

1. Règle de Bayes Nous obtenons

$$\tilde{\Xi}_{R_{12}} = \arg \max_{\Xi_{R_{12}}} [P' \cdot p(\Xi_{R_{12}})],$$

avec $p(\Xi_{R_{12}}) = p(\Xi_{\mathbf{a},R_{12}}) \cdot p(\Xi_{\mathbf{l},R_{12}})$. La distribution $p(\Xi_{\mathbf{a},R_{12}})$ correspondante à \mathbf{a} est donnée par l'Équation (E.4), celle correspondante à \mathbf{l} est introduite ci-dessous.

2. Distribution a priori des paramètres La distribution a priori sur les coefficients quadratiques \mathbf{l} est choisie comme

$$p(\Xi_{\mathbf{l},R_{12}}) = (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_{\mathbf{l}})^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{l}'^T \mathbf{R}_{\mathbf{l}}^{-1} \mathbf{l}'}, \quad (\text{E.10})$$

avec la matrice de corrélation $\mathbf{R}_{\mathbf{l}}$ qui contient l'information sur le lisseur qui correspond à

$$\mathbf{R}_{\mathbf{l}}^{-1} = \frac{\lambda_2}{\sigma^2} \mathbf{\Lambda}'_k,$$

analogue au choix de la matrice $\mathbf{R}_{\mathbf{a}}$ dans l'Équation (E.5), et $\mathbf{l}' \in \mathbb{R}^{2K^2+K-1}$ qui égale $\mathbf{l} \in \mathbb{R}^{2K^2+K}$ sauf le premier élément l_{11} . En choisissant d'utiliser la matrice $\mathbf{\Lambda}'_k$ au lieu de $\mathbf{\Lambda}_k$ (et par conséquent de ne pas régulariser l'élément l_{11}), nous évitons que $\det \mathbf{R}_{\mathbf{l}}$ ne soit nul et avec lui la probabilité $M_{N,R_{12}}$.

3. Expression de la vraisemblance En utilisant les expressions pour P'_N (Éq. (B.2)), $p(\Xi_{\mathbf{a}, R_{12}})$ (Éq. (E.4)) et $p(\Xi_{\mathbf{l}, R_{12}})$ (Éq. (E.10)), nous pouvons écrire

$$\begin{aligned} & P'_N \cdot p(\Xi_{R_{12}}) \\ &= \left[\frac{1}{\sqrt{2\pi}\sigma} \right]^N \exp \left[-\frac{Q'_{\mathbb{T}}(\Theta)}{2\sigma^2} \right] \cdot (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_{\mathbf{a}})^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{a}^\top \mathbf{R}_{\mathbf{a}}^{-1} \mathbf{a}} (2\pi)^{-\frac{N}{2}} (\det \mathbf{R}_{\mathbf{l}})^{-\frac{1}{2}} e^{-\frac{1}{2} \mathbf{l}'^\top \mathbf{R}_{\mathbf{l}}^{-1} \mathbf{l}'} . \end{aligned}$$

La matrice Λ'_k (Eq. (E.9)) est diagonale. On calcule que

$$(\det \mathbf{R}_{\mathbf{l}})^{-\frac{1}{2}} = \sqrt{\det \mathbf{R}_{\mathbf{l}}^{-1}} = \sqrt{\frac{\lambda_2^{2K^2+K-1}}{\sigma^{2(2K^2+K-1)}} \det \Lambda'_k} = \frac{\lambda_2^{K^2+\frac{K}{2}-\frac{1}{2}}}{\sigma^{2K^2+K-1}} \sqrt{\det \Lambda'_k}.$$

Définissons $\Lambda'_k \triangleq \sqrt{\det \Lambda'_k}$. On obtient

$$P'_N \cdot p(\Xi_{R_{12}}) = (2\pi)^{-\frac{3N}{2}} \frac{\lambda_1^K \lambda_2^{K^2+\frac{K}{2}-\frac{1}{2}}}{\sigma^{N+2K^2+3K-1}} [(2K)!]^k \Lambda'_k e^{-\frac{1}{2\sigma^2} (Q'_{\mathbb{T}} + \lambda_1 \mathbf{a}^\top \Delta_k \mathbf{a} + \lambda_2 \mathbf{l}'^\top \Lambda'_k \mathbf{l}')} .$$

Nous définissons

$$Q_{\mathbb{T}, R_{12}} \triangleq Q'_{\mathbb{T}} + \lambda_1 \mathbf{a}^\top \Delta_k \mathbf{a} + \lambda_2 \mathbf{l}'^\top \Lambda'_k \mathbf{l}' .$$

E.2.4 Maximisation de la vraisemblance

Nous définissons l'anti-log-vraisemblance

$$\begin{aligned} L_{\mathbb{T}, R_{12}} &\triangleq -\ln(P'_N \cdot p(\Xi_{R_{12}})) \\ &= \frac{3N}{2} \ln 2\pi - K \ln \lambda_1 - \left(K^2 + \frac{K}{2} - \frac{1}{2} \right) \ln \lambda_2 + (N + 2K^2 + 3K - 1) \ln \sigma \\ &\quad - k \ln[(2K)!] - \ln \Lambda'_k + \frac{1}{2\sigma^2} Q_{\mathbb{T}, R_{12}} . \end{aligned}$$

Comme dans le cas des méthodes $(R_1)QPC$, nous obtenons les Équations (4.10) pour les paramètres $\Xi_{R_{12}}$ en annulant les dérivées de l'anti-log-vraisemblance selon les paramètres,

$$\frac{\partial L_{\mathbb{T}, R_{12}}}{\partial \Xi_{R_{12}}} = \mathbf{0}_{2K^2+3K+1} .$$

Dérivée par rapport à σ

$$\frac{\partial L_{\mathbb{T}, R_{12}}}{\partial \sigma} = \frac{N + 2K^2 + 3K - 1}{\sigma^3} \left(\sigma^2 - \frac{Q_{\mathbb{T}, R_{12}}}{N + 2K^2 + 3K - 1} \right) = 0,$$

donc

$$\sigma = \sqrt{\frac{Q_{\mathbb{T}, R_{12}}}{N + 2K^2 + 3K - 1}} .$$

Dérivée par rapport à a

$$\begin{aligned} \frac{\partial L_{\mathbb{T}, R_{12}}}{\partial a_p} &= \frac{1}{2\sigma^2} \frac{\partial Q_{\mathbb{T}, R_{12}}}{\partial a_p} \\ &= \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-p] \right] \\ &\quad + \frac{1}{\sigma^2} \lambda_1 \cdot p^{2k} a_p \\ &= \frac{1}{\sigma^2} \left[R_{0,p} - \sum_{k=1}^{2K} R_{p,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j} + \lambda_1 \cdot p^{2k} a_p \right] = 0, \end{aligned}$$

donc

$$R_{p,0} = \sum_{k=1}^{2K} R_{p,k} a_k - \lambda_1 \cdot p^{2k} a_p + \sum_{i=1}^{2K} \sum_{j=i}^{2K} S_{p,i,j} l_{i,j}, \forall p \in \llbracket 1, 2K \rrbracket.$$

Cette expression est la même que dans la méthode R_1 QPC.

Dérivée par rapport à l

$$\frac{\partial L_{\mathbb{T}, R_{12}}}{\partial l_{r,s}} = \frac{1}{2\sigma^2} \frac{\partial Q_{\mathbb{T}, R_{12}}}{\partial l_{r,s}},$$

en utilisant l'Équation (E.8),

$$\begin{aligned} \frac{\partial L_{\mathbb{T}, R_{12}}}{\partial l_{r,s}} &= \frac{1}{\sigma^2} \sum_{n \in \mathbb{T}} \left[\left(y[n] - \sum_{k=1}^{2K} a_k y[n-k] - \sum_{i=1}^{2K} \sum_{j=i}^{2K} l_{i,j} y[n-i] y[n-j] \right) y[n-r] y[n-s] \right] \\ &\quad + \frac{1}{\sigma^2} \lambda_2 \cdot \left(\sum_{k_1+k_2=k} (s-1)^{2k_1} (r-1)^{2k_2} \right) l_{r,s} \\ &= \frac{1}{\sigma^2} \left[S_{0,r,s} - \sum_{k=1}^{2K} S_{r,s,k} a_k - \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j} + \lambda_2 \left(\sum_{k_1+k_2=k} (s-1)^{2k_1} (r-1)^{2k_2} \right) l_{r,s} \right] \\ &= 0, \end{aligned}$$

donc

$$S_{r,s,0} = \sum_{k=1}^{2K} S_{r,s,k} a_k + \sum_{i=1}^{2K} \sum_{j=i}^{2K} T_{r,s,i,j} l_{i,j} - \lambda_2 \left(\sum_{k_1+k_2=k} (s-1)^{2k_1} (r-1)^{2k_2} \right) l_{r,s},$$

$$\forall r \in \llbracket 1, 2K \rrbracket, s \in \llbracket r, 2K \rrbracket.$$

Dans cette équation un terme qui contient l'hyperparamètre λ_2 est rajouté, par rapport à la méthode R_1 QPC.

E.2.5 Estimation des hyperparamètres λ_1 et λ_2

Les valeurs des hyperparamètres λ_1 et λ_2 sont obtenues par la GCV, comme expliqué dans la Sous-section E.1.5 :

$$(\tilde{\lambda}_1, \tilde{\lambda}_2) = \arg \min \text{GCV}'(\lambda_1, \lambda_2), \text{ avec } \text{GCV}(\lambda_1, \lambda_2) = \frac{\|(\mathbf{I} - \mathbf{M}')\bar{\mathbf{y}}\|_2}{\text{Tr}(\mathbf{I} - \mathbf{M}')}$$

et ici

$$M' \triangleq \begin{bmatrix} \bar{Y}_1 & | & \bar{Y}_2 \end{bmatrix} \cdot (\mathbf{G}_{R_{12}} - \lambda_1 \mathbf{D}_1 - \lambda_2 \mathbf{D}_2)^{-1} \cdot \begin{bmatrix} \bar{Y}_1^\top \\ \bar{Y}_2^\top \end{bmatrix}.$$

Implémentation pratique Réécrivons M' comme

$$M' = \bar{Y} \mathbf{G}_{\lambda, R_{12}}^{-1} \bar{Y}^\top.$$

On implémente la GCV par l'expression

$$\text{GCV}(\lambda_1, \lambda_2) = \frac{\sqrt{\bar{y}^\top \bar{y} - 2(\bar{y}^\top \bar{Y}) \mathbf{G}_{\lambda, R_{12}}^{-1} (\bar{Y}^\top \bar{y}) + (\bar{y}^\top \bar{Y}) \mathbf{G}_{\lambda, R_{12}}^{-1} (\bar{Y}^\top \bar{Y}) \mathbf{G}_{\lambda, R_{12}}^{-1} (\bar{Y}^\top \bar{y})}}{(N-1) - \text{Tr}(\mathbf{G}_{\lambda, R_{12}}^{-1} (\bar{Y}^\top \bar{Y}))}.$$

Bibliographie

- [1] Thomas KAILATH. *Linear systems*, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.
- [2] Lennart LJUNG. *System Identification*. Linköping University Electronic Press, 2007.
- [3] Robert M. GRAY. *Linear Predictive Coding and the Internet Protocol : A survey of LPC and a History of Real time Digital Speech on Packet Networks*. Now Publishers, 2010.
- [4] Henrik HAHN. *Expressive Sampling Synthesis – Learning Extended Source–Filter Models from Instrument Sound Databases for Expressive Sample Manipulations*. Thèse de doctorat, UPMC Université Paris VI, 2015.
- [5] Hassan K. KHALIL. *Nonlinear systems*. Pearson, 2014.
- [6] Wilson J. RUGH. *Nonlinear system theory : the Volterra/Wiener approach*, 1981.
- [7] Damien BOUVIER, Thomas HÉLIE et David ROZE. Représentation en séries de Volterra d’un modèle passif de haut-parleur électrodynamique avec suspension non-linéaire et perspectives pour identification. *In Congrès Français d’Acoustique*, 2016.
- [8] Thomas HÉLIE. Modélisation physique d’instruments de musique et de la voix : Systèmes dynamiques, problèmes directs et inverses (mémoire pour obtenir l’habilitation à diriger des recherches), 2013.
- [9] Thomas HÉLIE. On the use of Volterra series for real-time simulations of weakly nonlinear analog audio devices : Application to the Moog ladder filter. *In Proceedings of the 9th International Conference on Digital Audio Effects (DAFx’06)*, 7–12. Citeseer, 2006.
- [10] Thomas HÉLIE. Volterra series and state transformation for real-time simulations of audio circuits including saturations : Application to the Moog ladder filter. *IEEE transactions on audio, speech, and language processing*, 18(4), 747–759, 2010.
- [11] Stephen A. BILLINGS. *Nonlinear system identification : NARMAX methods in the time, frequency, and spatio-temporal domains*. John Wiley & Sons, 2013.
- [12] Angelo FARINA. Simultaneous measurement of impulse response and distortion with a swept-sine technique. *In Audio Engineering Society Convention 108*. Audio Engineering Society, 2000.
- [13] Antonín NOVÁK, Laurent SIMON, František KADLEC et Pierrick LOTTON. Nonlinear system identification using exponential swept-sine signal. *IEEE Transactions on Instrumentation and Measurement*, 59(8), 2220–2229, 2010.

- [14] Marc RÉBILLAT, Romain HENNEQUIN, Etienne CORTEEL et Brian F. G. KATZ. Identification of cascade of Hammerstein models for the description of nonlinearities in vibrating devices. *Journal of Sound and Vibration*, 330(5), 1018–1038, 2011.
- [15] Gunnar FANT. *Acoustic Theory of Speech Production*. Mouton, The Hague, 1970.
- [16] Thomas HÉLIE. Vocoder par LPC. In Brigitte D’ANDRÉA-NOVEL, Benoît FABRE et Pierre JOUVELOT, éditeurs. *Acoustique Informatique MusiquE*, chapitre 11. Transvalor – Presses des Mine, 2012.
- [17] John D. MARKEL et Augustine H. GRAY JR. *Linear prediction of speech*, volume 12. Springer Science & Business Media, 2013.
- [18] Marco C. CAMPI, Balázs C. CSÁJI, Simone GARATTI et Erik WEYER. Certified system identification - towards distribution-free results. *IFAC Proceedings Volumes*, 45(16), 245–255, 2012.
- [19] Genshiro KITAGAWA et Will GERSCH. A smoothness priors long AR model method for spectral estimation. *IEEE transactions on automatic control*, 30(1), 57–65, 1985.
- [20] Jean-François GIOVANNELLI, Guy DEMOMENT et Alain HERMENT. A Bayesian method for long AR spectral estimation : A comparative study. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 43(2), 220–233, 1996.
- [21] Stefan BILBAO. *Numerical sound synthesis : finite difference schemes and simulation in musical acoustics*. John Wiley & Sons, 2009.
- [22] Thomas HÉLIE et Béatrice LAROCHE. On the convergence of Volterra series of finite dimensional quadratic MIMO systems. *International Journal of Control, special issue in Honor of Michel Fliess 60th-birthday*, 81(3), 358–370, 2008.
- [23] Thomas HÉLIE et David ROZE. Sound synthesis of a nonlinear string using Volterra series. *Journal of Sound and Vibration*, 314(1), 275–306, 2008.
- [24] Thomas HÉLIE et Vanessa SMET. Simulation of the weakly nonlinear propagation in a straight pipe : application to a real-time brassy audio effect. In *Control and Automation, 2008 16th Mediterranean Conference on*, 1580–1585. IEEE, 2008.
- [25] Jean-François GIOVANNELLI. *Estimation de caractéristiques spectrales en temps court. Application à l’imagerie Doppler*. Thèse de doctorat, Laboratoire des Signaux et Systèmes, Université Paris-Sud, Orsay, France, 1995.
- [26] Gene H. GOLUB, Michael HEATH et Grace WAHBA. Generalized cross-validation as a method for choosing a good ridge parameter. *TECHNOMETRICS*, 21(2), 1979.
- [27] VIENNA SYMPHONIC LIBRARY GMBH. Brass & Woodwinds – First Edition. 2003. URL : www.vsl.co.at.
- [28] Hirotugu AKAIKE. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6), 716–723, 1974.
- [29] Michel KIEFFER et Eric WALTER. Guaranteed characterization of exact non-asymptotic confidence regions as defined by LSCR and SPS. *Automatica*, 50(2), 507–512, 2014.

-
- [30] Marco C. CAMPI et Erik WEYER. Guaranteed non-asymptotic confidence regions in system identification. *Automatica*, 41(10), 1751–1764, 2005.
- [31] Marco DALAI, Erik WEYER et Marco C. CAMPI. Parameter identification for nonlinear systems : Guaranteed confidence regions through LSCR. *Automatica*, 43(8), 1418–1425, 2007.
- [32] Balázs Csanád CSÁJI, Marco C. CAMPI et Erik WEYER. Non-asymptotic confidence regions for the least-squares estimate. *IFAC Proceedings Volumes*, 45(16), 227–232, 2012.
- [33] Balázs Csanád CSÁJI, Marco C. CAMPI et Erik WEYER. Strong consistency of the sign-perturbed sums method. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, 3352–3357. IEEE, 2014.
- [34] Balázs Csanád CSÁJI, Marco Claudio CAMPI et Erik WEYER. Sign-perturbed sums : A new system identification approach for constructing exact non-asymptotic confidence regions in linear regression models. *IEEE Transactions on Signal Processing*, 63(1), 169–181, 2015.
- [35] Ronald Aylmer FISHER. Theory of statistical estimation. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 22, 700–725. Cambridge Univ Press, 1925.
- [36] Calyampudi Radhakrishna RAO. Information and the accuracy attainable in the estimation of statistical parameters. 235–247. 1945.
- [37] Harald CRAMER. Mathematical Methods of Statistics (Princeton : Princeton University Press, 1946). *Mathematical Methods of Statistics*, 1946.

Liste des figures

1.1	Modélisation en source et système dynamique d'un instrument de musique	2
1.2	Deux catégories d'identification	4
1.3	Principe de la LPC	5
2.1	Réalisation des noyaux de Volterra des trois premiers ordres [8]	12
3.1	Représentation des vecteurs \mathbf{a} et \mathbf{l} du système simulé.	16
3.2	Estimation du système dans le cas $\sigma = 0,5$: représentation graphique des estimations des vecteurs \mathbf{a} et \mathbf{l} . Les lignes tiretées indiquent les valeurs exactes \mathbf{a} et \mathbf{l} , les lignes pleines leurs estimations $\tilde{\mathbf{a}}$ et $\tilde{\mathbf{l}}$	18
4.1	Signification du facteur $R_k^{(1)}$: le spectre en ligne pleine est moins lisse que le spectre en ligne tiretée donc son facteur $R_k^{(1)}$ est plus grand.	22
4.2	Estimation $\tilde{\lambda}_1$ pour une simulation avec $N = 10^5$, $\sigma = 1$. Pour des raisons numériques, dans la figure, la valeur de la GCV est normalisée en soustrayant le minimum g atteint sur l'intervalle considéré. Le minimum g de la GCV est obtenu pour $\tilde{\lambda}_1 = -1,500$	26

Liste des tableaux

3.1	Estimation du système dans le cas $\sigma = 0,5$: évolution des erreurs en fonction du nombre de données N	17
3.2	Estimation du système : évolution des erreurs en fonction de l'écart type σ et du nombre de données N	19
4.1	Erreurs sur l'estimation des paramètres par les méthodes QPC et R ₁ QPC pour un même signal d'entrée avec $N = 10^5$, $\sigma = 1$, correspondant à la Figure 4.2.	27

4.2 Erreurs sur l'estimation des paramètres par les méthodes QPC et R₁QPC (en fonction de valeurs positives de λ_1) pour un même signal d'entrée avec $N = 10^5, \sigma = 1$ 27