Représentations parcimonieuses pour les signaux multivariés

Quentin Barthélemy

▶ **To cite this version:**

Quentin Barthélemy. Représentations parcimonieuses pour les signaux multivariés. Sciences de l'ingénieur [physics]. Université de Grenoble, 2013. Français. NNT : . tel-00853362v1

HAL Id: tel-00853362

https://theses.hal.science/tel-00853362v1

Submitted on 22 Aug 2013 (v1), last revised 22 Jan 2014 (v2)

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

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

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Sciences de la Terre et de l'Univers et de l'Environnement**

Arrêté ministériel : 7 août 2006

Présentée par

Quentin BARTHÉLEMY

Thèse dirigée par **Jérôme I. MARS**
et encadrée par **Anthony LARUE**

préparée aux laboratoires **GIPSA-Lab et CEA-LIST**
dans l'**Ecole Doctorale Terre, Univers, Environnement**

Représentations parcimonieuses pour les signaux multivariés

Thèse soutenue publiquement le **13 Mai 2013**,
devant le jury composé de :

Monsieur Christian JUTTEN
PR, Université de Grenoble, Président
Monsieur Rémi GRIBONVAL
DR, INRIA, Rapporteur
Monsieur Alain RAKOTOMAMONJY
PR, Université de Rouen, Rapporteur
Monsieur Yanis CARITU
Ing., Movea, Examinateur
Monsieur Jérôme I. MARS
PR, Grenoble-INP, Directeur de thèse
Monsieur Anthony LARUE
Ing., CEA, Encadrant de thèse

à Marie-Victoire, ma femme

Représentations parcimonieuses pour les signaux multivariés

Remerciements

Je remercie d'abord les membres du jury de thèse pour le temps qu'ils ont consacré à l'examen de mon manuscrit et pour leurs remarques.

Je remercie Jérôme Mars, mon directeur de thèse, pour son aide précieuse tout au long de cette aventure commencée au GIPSA-Lab en juin 2009.

Je remercie tout spécialement Anthony Larue, mon encadrant de thèse, toujours disponible malgré ses responsabilités sans cesse grandissantes. Ma thèse n'aurait pas été la même sans ses avis toujours pertinents et ses nombreuses heures de relecture d'articles.

Je remercie aussi toutes les personnes du laboratoire, pour leurs conseils ainsi que ces quelques mois passés ensembles.

Je remercie enfin mes amis et ma famille pour leurs aides et leurs amitiés.

Mes remerciements les plus chers vont bien évidemment à ma femme, pour son soutien constant, notamment lors de la période de rédaction.

Table des matières

Table des matières ix

Notations xiii

Abréviations xvi

Introduction 1

1 Représentations parcimonieuses 3

Introduction . . . 3

1.1 Décomposition d'un signal sur un dictionnaire . . . 3

1.1.1 Décomposition linéaire . . . 3

1.1.2 Analyse en composantes . . . 4

1.1.3 Cas particulier de redondance : l'invariance par translation . . . 6

1.2 Approximation parcimonieuse . . . 8

1.2.1 Formulations de l'approximation parcimonieuse . . . 8

1.2.2 Présentation MP et OMP . . . 10

1.2.3 OMP dans le cas d'invariance par translation . . . 11

1.2.4 Autres algorithmes d'approximation parcimonieuse . . . 14

1.3 Apprentissage de dictionnaire . . . 15

1.3.1 Principe de l'apprentissage de dictionnaire . . . 16

1.3.2 Revue des méthodes d'apprentissage . . . 17

1.3.3 Remarques . . . 21

1.3.4 DLA invariants par translation . . . 21

1.4 Considérations théoriques sur la cohérence . . . 22

1.4.1 Cas classique . . . 22

1.4.2 Cas invariant par translation . . . 23

Conclusion . . . 28

2 Représentations parcimonieuses multivariées 37

Introduction . . . 37

2.1 Modèle multicanal versus multivarié . . . 37
2.1.1 Présentation comparative des modèles . . . 38
2.1.2 Apprentissages de dictionnaires multicomposantes . . . 39
2.2 OMP multivarié . . . 40
2.2.1 Description du *Multivariate*-OMP . . . 40
2.2.2 Remarques sur le modèle multivarié . . . 42
2.3 DLA multivarié . . . 43
2.3.1 Description du *Multivariate*-DLA . . . 43
2.3.2 Remarques sur le *Multivariate*-DLA . . . 44
2.4 Comparaison des méthodes sur des données simulées . . . 45
2.5 Applications aux signaux électroencéphalographiques . . . 46
2.5.1 Etat de l'art de l'analyse temps-fréquence pour les EEG . . . 46
2.5.2 Expérience 1 : apprentissage et décompositions . . . 48
2.5.3 Experience 2 : apprentissage de potentiels évoqués . . . 53
Conclusion . . . 60

3 Représentations parcimonieuses quaternioniques 67
Introduction . . . 67
3.1 Modèles linéaires quaternioniques . . . 67
3.1.1 Algèbre des quaternions . . . 67
3.1.2 Modèle linéaire avec multiplication à droite . . . 68
3.1.3 Modèle linéaire avec multiplication à gauche . . . 68
3.2 OMP quaternioniques . . . 69
3.2.1 Description du Q-OMP à gauche (*resp.* à droite) . . . 69
3.2.2 Extension des Q-OMP au cas d'invariance par translation . . . 71
3.2.3 Spikegramme . . . 72
3.3 Expériences sur données simulées . . . 73
3.3.1 Débruitage et déconvolution . . . 73
3.3.2 Comparaisons . . . 75
3.3.3 Complexité . . . 76
Conclusion . . . 76

4 Représentations invariantes par rotation 2D 79
Introduction . . . 79
4.1 Invariance par rotation 2D . . . 79
4.1.1 Formulation du problème . . . 79
4.1.2 Méthodes d'apprentissage de primitives du mouvement . . . 80
4.2 Méthodes 2DRI . . . 81
4.2.1 Approche 2DRI . . . 81

4.2.2 Description du 2DRI-OMP et du 2DRI-DLA . . . 82
4.2.3 Remarques sur le 2DRI-OMP . . . 83
4.3 Application aux données d'écriture manuscrite . . . 83
4.3.1 Données de mouvements 2D . . . 83
4.3.2 Expérience 1 : apprentissage de dictionnaires . . . 84
4.3.3 Expérience 2 : décompositions des données . . . 88
4.3.4 Expérience 3 : décompositions des données tournées . . . 89
4.4 Comparaisons . . . 91
4.4.1 Comparaison avec des dictionnaires classiques . . . 91
4.4.2 Comparaison entre les apprentissages orientés et non-orientés . . . 92
4.5 Discussions . . . 94
4.5.1 Considérations sur le modèle 2DRI . . . 94
4.5.2 Limitations de la méthode . . . 95
Conclusion . . . 96

5 Représentations invariantes par rotation 3D **99**
Introduction . . . 99
5.1 Invariance par rotation 3D . . . 99
5.2 Etat de l'art des décompositions 3D . . . 100
5.2.1 Modèles de décompositions tricomposantes . . . 100
5.2.2 Problèmes de minimisation 3D . . . 102
5.2.3 Problème complet . . . 103
5.3 MP invariant par rotation 3D . . . 104
5.3.1 Recalage 3D par SVD . . . 104
5.3.2 Description du 3DRI-MP . . . 105
5.3.3 Remarques sur le 3DRI-MP . . . 105
5.4 OMP invariant par rotation 3D . . . 106
5.4.1 Description du 3DRI-OMP . . . 107
5.4.2 Procédure d'optimisation pour les coefficients et les matrices . . . 107
5.4.3 Remarques sur le 3DRI-OMP . . . 109
5.5 Etat de l'art des apprentissages 3D . . . 109
5.5.1 Apprentissage de base . . . 109
5.5.2 Apprentissage de dictionnaire . . . 110
5.6 DLA invariant par rotation 3D . . . 110
5.6.1 Description du 3DRI-DLA . . . 110
5.6.2 Remarques sur le 3DRI-DLA . . . 111
5.7 Expériences sur données simulées . . . 111
5.7.1 Décompositions invariantes par rotation 3D . . . 112

5.7.2 Apprentissages de dictionnaires invariants par rotation 3D . . . 113
5.8 Expériences sur données réelles . . . 114
5.8.1 Données d'application . . . 114
5.8.2 Expérience 1 : apprentissage de dictionnaires . . . 117
5.8.3 Expérience 2 : décompositions des données . . . 118
5.8.4 Expérience 3 : décompositions des données tournées . . . 121
5.9 Extensions . . . 123
5.9.1 Extension nDRI . . . 123
5.9.2 Extension multicapteurs . . . 124
Conclusion . . . 125

6 Classification invariante par translation **129**
Introduction . . . 129
6.1 Descripteurs consistants par translation . . . 129
6.1.1 Formulation du problème . . . 130
6.1.2 Etat de l'art sur les fonctions de groupement . . . 130
6.2 Comparaisons des différentes fonctions de groupement . . . 134
6.2.1 Apprentissages des classifieurs et comparaisons . . . 134
6.2.2 Tests de consistances . . . 136
6.3 Nouvelle famille de fonctions de goupement . . . 137
6.3.1 Nouveau fenêtrage . . . 137
6.3.2 Test de consistance . . . 138
6.3.3 Discussion . . . 138
6.4 Apprentissage de dictionnaire discriminant . . . 139
6.4.1 Contexte . . . 140
6.4.2 Cas classique : état de l'art . . . 140
6.4.3 Cas invariant par translation . . . 142
Conclusion . . . 145

Conclusion et perspectives **149**

7 Annexes **153**
7.1 Complément pour la Section 1.4.2 . . . 153
7.2 Considérations sur l'implémentation du M-OMP . . . 153
7.3 Calcul du Hessien . . . 154
7.4 Produits scalaires pour les quaternions . . . 155
7.5 Etape de sélection pour le Q-OMPr . . . 156
7.6 Etape de sélection pour le Q-OMPl . . . 157
7.7 Problème de Procrustes orthogonal . . . 158

Bibliographie générale **177**

Publications **179**

Notations

Ensembles

$\mathbb{N}_N$: ensemble des nombres entiers naturels allant de 1 à N
$\mathbb{R}$: ensemble des nombres réels
$\mathbb{C}$: ensemble des nombres complexes
$\mathbb{H}$: ensemble des quaternions
i : imaginaire pur des complexes $\mathbb{C}$ et des quaternions $\mathbb{H}$
j et k : imaginaires purs des quaternions $\mathbb{H}$
$\Re$: partie réelle pour les complexes
$\Im$: partie imaginaire pour les complexes
$\mathcal{S}$: partie scalaire pour les quaternions
$\mathcal{V}$: partie vectorielle pour les quaternions

Opérateurs

$(.)^T$: opérateur transposé
$(.)^*$: opérateur conjugué
$(.)^H$: opérateur transconjugué
$(.)^\dagger$: opérateur pseudo-inverse
$\mathrm{Tr}(.)$: opérateur trace
$\langle\ ,\ \rangle$: produit scalaire, redéfini dans chaque chapitre
$\|.\|_p$: norme ℓ_p
$\|.\|_0$: pseudo-norme ℓ_0
$\|.\|_F$: norme de Frobenius
$[\ ;\ ]$: concaténation verticale (selon la $1^{\text{ère}}$ dimension)
$[\ ,\ ]$: concaténation horizontale (selon la $2^{\text{ième}}$ dimension)
$\left(\ ;\ \right)$: concaténation en profondeur (selon la $3^{\text{ième}}$ dimension)

Indices

N : nombre d'échantillons temporels, indicés par t
M : nombre d'atomes, indicés par m
L : nombre de noyaux, indicés par l

T_l : nombre d'échantillons du noyau l
V : nombre de composantes (canaux, variables, modalités), indicées par v
P : nombre de signaux, indicés par p
Q : nombre de signaux de test, indicés par q
K : parcimonie, *i.e.* nombre d'éléments non nuls, indicée par k

Signaux

$y \in \mathbb{R}^N$: signal (par défaut : unicomposante, monocomposante, univarié)
$\hat{y}$: approximation du signal y
$\boldsymbol{y} \in \mathbb{R}^{N \times V}$: signal avec V composantes (multicanal ou multivarié, selon le modèle utilisé)
$\boldsymbol{y}[v]$: $v$ième composante du signal $\boldsymbol{y}$
$\Phi \in \mathbb{R}^{N \times M}$: dictionnaire d'atomes $\left[\phi_m \in \mathbb{R}^N\right]_{m=1}^M$
$\boldsymbol{\Phi} \in \mathbb{R}^{N \times M \times V}$: dictionnaire d'atomes multivariés $\left[\boldsymbol{\phi}_m \in \mathbb{R}^{N \times V}\right]_{m=1}^M$
Ψ : dictionnaire de noyaux $\left\{\psi_l \in \mathbb{R}^{T_l}\right\}_{l=1}^L$
$\boldsymbol{\Psi}$: dictionnaire de noyaux multivariés $\left\{\boldsymbol{\psi}_l \in \mathbb{R}^{T_l \times V}\right\}_{l=1}^L$
$x \in \mathbb{R}^M$: vecteur de coefficients
$\boldsymbol{x} \in \mathbb{R}^{M \times V}$: vecteur de coefficients multicanaux
$\epsilon \in \mathbb{R}^N$: erreur résiduelle
$\boldsymbol{\epsilon} \in \mathbb{R}^{N \times V}$: erreur résiduelle multicomposante associée au signal $\boldsymbol{y}$

Bases de données

Y : base de données composée de P signaux
$\boldsymbol{Y}$: base de données composée de P signaux multicomposantes
X : vecteurs de coefficients
$\boldsymbol{X}$: vecteurs de coefficients multicanaux
E : erreurs résiduelles associées à la base Y
$\boldsymbol{E}$: erreurs résiduelles multicomposantes associées à la base $\boldsymbol{Y}$

Autres

$\mathfrak{p}(.)$: probabilité
$\Gamma(t)$: fonction de corrélation

Abréviations

Abréviations françaises

EEG : Electroencéphalogramme
ICM : Interface Cerveau-Machine
LPC : Langage Parlé Complété
RSB : Rapport Signal sur Bruit

Abréviations anglaises

Acronymes :
BCI : *Brain-Computer Interface*
DCT : *Discrete Cosine Transform*
DFT : *Discrete Fourier Transform*
DLA : *Dictionary Learning Algorithm*
EM : *Expectation-Maximization*
FFT : *Fast Fourier Transform*
ICA : *Independent Component Analysis*
LMS : *Least Mean Squares*
MP : *Matching Pursuit*
MSE : *Mean Square Error*
NOLD : *Non-Oriented Learned Dictionary*
OLD : *Oriented Learned Dictionary*
OMP : *Orthogonal Matching Pursuit*
PCA : *Principal Component Analysis*
RMSE : *Root Mean Square Error*
rRMSE : *relative Root Mean Square Error*
SCA : *Sparse Component Analysis*
SVD : *Singular Value Decomposition*
WT : *Wavelet Transform*

Préfixes :
M : *Multivariate*
Mch : *Multichannel*
2DRI : *2D Rotation Invariant*
3DRI : *3D Rotation Invariant*
nDRI : *nD Rotation Invariant*

Introduction

Depuis quelques années, la performance des systèmes d'acquisition et la taille grandissante des mémoires de stockage ont favorisé la construction de grandes bases de données, composées d'une collection de signaux. Au vu de leurs tailles, on parle même de masses de données et leur exploitation postérieure devient un problème crucial. Un des enjeux est de trouver l'information utile au sein d'une masse de données.

En exploration et analyse de données, il existe plusieurs approches pour traiter ce problème. Les données acquises vivent souvent dans des espaces de grandes dimensions et sont souvent très redondantes. A l'inverse, l'information utile enfouie dans ces données vit bien souvent dans des sous-espaces de petites dimensions. Aussi, ces données peuvent être représentées avec parcimonie, *i.e.* avec peu d'éléments, à l'aide de composantes informatives sélectionnées parmi un ensemble de composantes apprises *ad hoc*.

Ainsi, le problème se divise donc en deux. Dans un premier temps, nous voulons apprendre ces composantes informatives, futurs supports des représentations. Il s'agit donc d'extraire sans *a priori* de la base de données les structures informatives, répétitives et énergétiques. A l'inverse des méthodes d'analyse en composantes habituelles qui fournissent une base de l'espace (analyses en composantes principales ou indépendantes), nous apprenons une base redondante de composantes appelée dictionnaire. Dans un deuxième temps, nous utilisons ces composantes apprises pour représenter chaque signal de la base de données. La représentation parcimonieuse résultante a plusieurs avantages :

- elle permet de représenter de façon optimale chaque signal grâce à un sous-ensemble spécifique des composantes redondantes disponibles, cette souplesse de sélection fournit une représentation adaptative ;
- elle permet de réduire la taille des données, avec un taux de fidélité/perte maîtrisable ;
- elle permet de présenter à un utilisateur les données étudiées, en ne gardant que l'information utile et ce, de manière condensée.

L'inconvénient des transformées classiques par rapport à l'analyse sur composantes apprises est qu'elles injectent des *a priori* via leurs composantes prédéfinies analytiquement.

Dans cette thèse, nous étudierons des données composées de signaux temporels multivariés. Ces signaux résultent de l'acquisition simultanée de plusieurs grandeurs, tels que des signaux électroencéphalographiques acquis par plusieurs électrodes ou des signaux de mouvements 2D et 3D. Au lieu de traiter indépendamment les signaux de chaque canal, ce qui est une perte d'information, l'approche multivariée que nous introduisons prend en compte les liens entre les différents canaux. Ces données de grandes dimensions s'avèrent être très redondantes car des structures élémentaires caractéristiques se répètent au sein des données, très souvent à un facteur d'échelle près, à une translation temporelle près, à une rotation près, etc. Cette redondance peut être surmontée au prix de l'ajout de degrés de liberté. Nous

mettons en place une modélisation des signaux qui est invariante à l'échelle, à la translation temporelle et à la rotation. Ainsi, les composantes de représentations, de petites dimensions et en nombre restreint, sont démultipliables en générant des répliques d'elles-mêmes à tous les facteurs d'échelle possibles, à toutes les translations temporelles et à toutes les rotations possibles.

Dans un premier chapitre d'introduction, nous présenterons un état de l'art sur les méthodes d'approximation parcimonieuse ℓ_0 et ℓ_1 et d'apprentissage de dictionnaire, fournissant des représentations adaptées parcimonieuses à une base de données. Le cas d'invariance par translation sera détaillé pour l'étude postérieure de signaux temporels. Nous exposerons ensuite l'extension de ces méthodes au modèle multivarié en Chapitre 2, en précisant la différence avec le modèle multicanal classiquement utilisé. Nous en ferons l'illustration sur des données électroencéphalographiques en apprenant une représentation efficace et ayant une interprétation physiologique. En Chapitre 3, ces méthodes seront étudiées dans le cadre particulier de l'algèbre des quaternions qui permet de traiter des données quadrivariées. A cause de la non-commutativité des quaternions, deux modèles linéaires seront proposés en fonction de l'ordre de la multiplication. Dans les Chapitres 4 et 5, nous détaillerons comment obtenir des représentations parcimonieuses invariantes par rotation 2D et 3D. Les méthodes de représentations parcimonieuses seront donc spécifiées pour résoudre ces deux problèmes. Ces méthodes seront ensuite illustrées sur des signaux d'écriture manuscrite en 2D et des signaux de Langage Parlé Complété en 3D. Si les représentations invariantes sont utiles pour surmonter la redondance des données, elles sont indispensables pour des tâches de haut niveau comme la classification. Le Chapitre 6 s'intéressera à la classification adaptée à de telles représentations et à l'amélioration possible de cette classification.

Chapitre 1

Représentations parcimonieuses

Introduction

Choisir la bonne représentation des données est devenu un problème crucial au vu de la taille grandissante des bases de données, acquises dans des espaces de grandes dimensions. L'information utile enfouie dans ces données vit bien souvent dans des sous-espaces de petites dimensions. Les méthodes de représentations adaptatives, alliant redondance et parcimonie, fournissent cette représentation efficace.

Dans ce chapitre, nous ferons l'état de l'art méthodologique sur les représentations parcimonieuses. Nous expliquerons la décomposition d'un signal sur un dictionnaire redondant, puis la façon d'estimer les coefficients de décomposition par approximation parcimonieuse, et enfin l'apprentissage de dictionnaire qui permet d'inférer le dictionnaire le mieux adapté aux données.

1.1 Décomposition d'un signal sur un dictionnaire

Dans cette section, nous considérons des signaux issus d'une base de données et nous nous intéressons à la manière de les analyser de façon pertinente.

1.1.1 Décomposition linéaire

Pour traiter des signaux dans un espace de Hilbert, nous définissons le produit scalaire entre deux signaux a et $b \in \mathbb{R}^N$ comme $\langle a, b\rangle = b^T a$, avec $(.)^T$ représentant l'opérateur transposé. La norme ℓ_2 associée est notée $\|.\|_2$, et plus généralement la norme ℓ_s est notée $\|a\|_s^s = \sum_{n=1}^N |a_n|^s$. Dans le cas matriciel, $\langle A, B\rangle = \mathrm{Tr}(B^T A)$, avec la norme de Frobenius associée notée $\|.\|_F$.

Nous considérons un signal $y \in \mathbb{R}^N$ composé de N échantillons temporels et une matrice $\Phi \in \mathbb{R}^{N\times M}$ composée de M colonnes appelées atomes $\{\phi_m\}_{m=1}^M$ (ou encore vecteurs, fonctions). La décomposition linéaire du signal y sur Φ s'écrit :

$$y = \Phi x + \epsilon \,, \tag{1.1}$$

$$= \sum_{m=1}^{M} \phi_m x_m + \epsilon \,, \tag{1.2}$$

avec $x \in \mathbb{R}^M$ le vecteur de coefficients (ou d'activations), et $\epsilon \in \mathbb{R}^N$ l'erreur résiduelle. Dans ce modèle, le signal y est la combinaison linéaire (*i.e.* la somme pondérée) des atomes de Φ. L'approximation de y est notée $\hat{y} = \Phi x$.

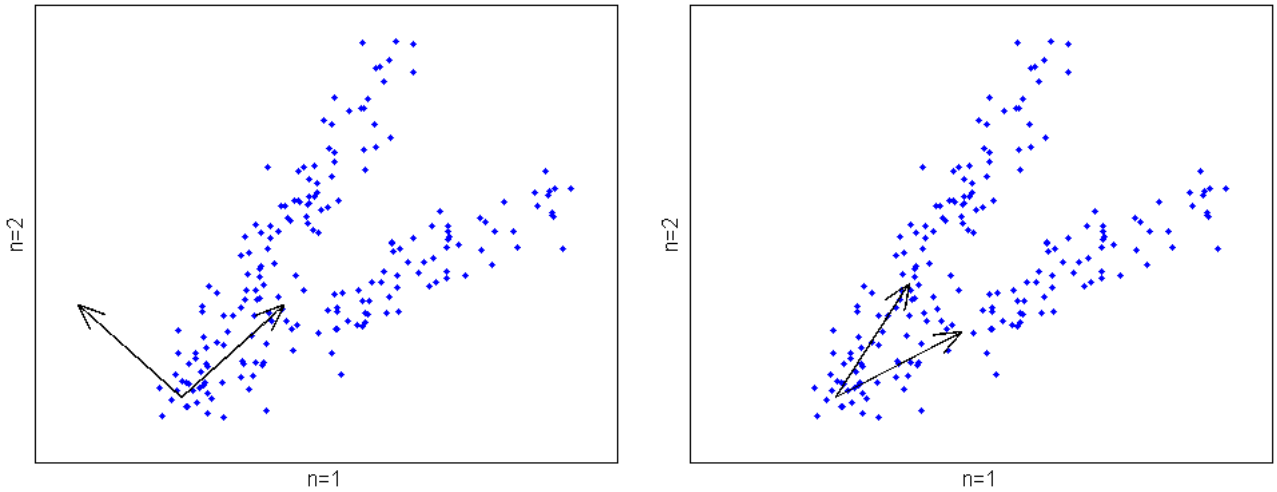
Nous définissons quelques outils utiles pour la suite. Le support du vecteur x est défini par : $\text{support}(x) = \{m \in \mathbb{N}_M : x_m \neq 0\}$. Le support résume donc les coefficients non nuls de x. La cohérence mutuelle d'un dictionnaire Φ est définie comme la valeur absolue maximale des produits scalaires entre atomes : $\mu_\Phi = \max_{i \neq j} \left| \langle \phi_i, \phi_j \rangle / (\|\phi_i\|_2 \|\phi_j\|_2) \right|$. Cette valeur sert à caractériser le dictionnaire comme nous le verrons ultérieurement. Par la suite, le dictionnaire sera généralement considéré normé, *i.e.* chaque atome est de norme égale à 1.

1.1.2 Analyse en composantes

Différentes façons existent pour analyser un ensemble de données composé de P signaux. Ces techniques sont regroupées sous le nom d'Analyse en Composantes, où composantes est synonyme d'atomes.

Analyse en Composantes Principales

L'Analyse en Composantes Principales [CJ10], en anglais *Principal Component Analysis* (PCA), est très employée pour obtenir une base adaptée apprise à partir des données, et faire de la réduction de dimension. Les vecteurs appris sont appelés composantes principales et expliquent le maximum de variance des données. Ce sont les vecteurs propres de la matrice de covariance centrée, et ils sont orthogonaux. L'inconvénient majeur est que la PCA met un *a priori* Gaussien sur la densité de distribution des données.



(a) Composantes de la PCA.

(b) Composantes de l'ICA.

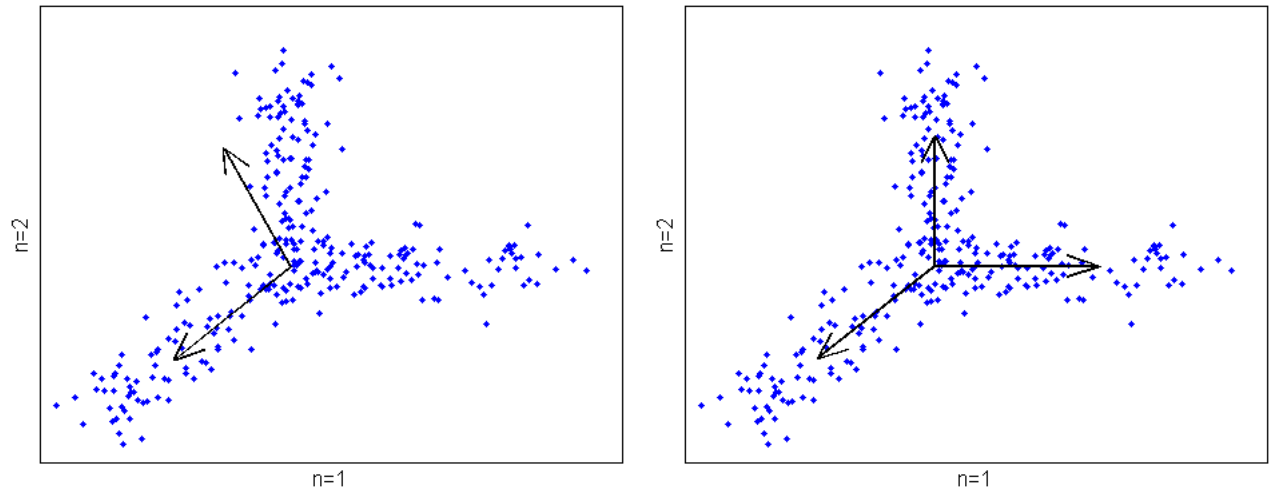
Fig. 1.1 – Illustration de données distribuées en V (distribution non Gaussienne) dans le cas $N = 2$ (points bleus) et les composantes apprises par la PCA (a) et par l'ICA (b) (vecteurs noirs).

Analyse en Composantes Indépendantes

L'Analyse en Composantes Indépendantes [Com94, CJ10], en anglais *Independent Component Analysis* (ICA), permet d'obtenir des composantes non-orthogonales, ce qui est plus intéressant quand les données ne sont pas distribuées de façon Gaussienne. Elle est très employée quand les signaux sont mutuellement indépendants, comme dans la séparation de sources. Elle a été appliquée à de multiples types de signaux [HO00], comme les signaux électroencéphalographiques [JMH+00], électrocardiographiques [SJS08] ou les ondes sismiques [VML04].

La Fig. 1.1 montre un exemple de données de distribution non Gaussienne (en V), avec $P = 200$ signaux $y \in \mathbb{R}^N$ et avec $N = 2$. Les vecteurs de la PCA pointent dans des directions sur lesquelles il n'y a que très peu de signaux, ce qui signifie que la distribution des données est mal estimée. Au contraire, les vecteurs de l'ICA pointent sur les régions de hautes densités de l'espace de données. Dans ces deux cas, le nombre de vecteurs obtenus M est égal à la dimension N des signaux. Les vecteurs forment donc une base $\Phi \in \mathbb{R}^{N \times N}$ qui génère l'espace $\mathbb{R}^N$, ce qui est suffisant pour représenter les données. Mais cette propriété possède aussi des inconvénients.

La Fig. 1.2 donne un exemple de cette limitation avec des données distribuées de façon tri-axiale, avec $P = 300$ signaux et $N = 2$. Dans ce cas, les vecteurs de l'ICA ne sont pas pertinents : même s'il est possible de représenter les données par une combinaison linéaire des $M = 2$ vecteurs, cette représentation n'est pas simple du point de vue des coefficients. De plus, les vecteurs souffrent d'un manque de signification par rapport aux données.



(a) Composantes de l'ICA.

(b) Composantes de la SCA.

Fig. 1.2 – Illustration de données distribuées de façon tri-axiale dans le cas $N = 2$ (points bleus) et les composantes apprises par l'ICA (a) et par la SCA (b) (vecteurs noirs).

Analyse en Composantes Parcimonieuses

L'Analyse en Composantes Parcimonieuses [LS98], en anglais *Sparse Component Analysis* (SCA), repose sur deux principes : la redondance et la parcimonie. D'une part, la matrice Φ contient plus de composantes que la taille de l'espace, *i.e.* $M \geq N$. Cette base redondante $\Phi \in \mathbb{R}^{N \times M}$ est appelée

dictionnaire [Mal09]. L'avantage de cette redondance est la flexibilité et la finesse des représentations issues de ce dictionnaire : plusieurs vecteurs proches pourront décrire les nuances d'un même phénomène. L'inconvénient est que plusieurs solutions sont possibles pour résoudre le système sous-déterminé (1.2). D'autre part, la parcimonie (*sparsity* en anglais) consiste à expliquer un phénomène avec le minimum de causes élémentaires. Elle a pour conséquence de sélectionner les éléments les plus pertinents parmi une multitude. Ainsi, dans la SCA, la parcimonie et la redondance sont complémentaires. L'alliance des deux assure une représentation simple et efficace : la parcimonie sélectionne les vecteurs optimaux parmi le dictionnaire pour avoir la meilleure représentation. Au final, l'information utile est condensée en quelques éléments choisis *ad hoc*, de façon à ce que la représentation soit à la fois compacte et fidèle aux données.

Sur la Fig. 1.2, grâce à sa redondance, la SCA fournit $M = 3$ composantes pour représenter efficacement les données. En fonction du signal à analyser, la parcimonie choisira un seul des trois vecteurs du dictionnaire : chaque signal sera donc toujours décomposé sur un seul vecteur. La flexibilité de la représentation lui confère sa simplicité et donc sa pertinence : on parle ainsi de représentation adaptative.

Nous pouvons établir une comparaison entre un dictionnaire d'atomes et une caisse à outils. Si la caisse est redondante, nous aurons à disposition de multiples outils : plusieurs tournevis, des clés Allen, des clés plates de différentes tailles, plusieurs pinces, etc. Pour serrer un écrou de 9, nous trouverons dans la caisse une clé adéquate ; s'il faut en serrer un de 10, nous prendrons la clé de la taille au-dessus. En revanche, si la caisse comporte peu d'outils et/ou mal adaptés, nous serons contraints de serrer l'écrou de 9 avec une clé de 10, voire avec une pince. Pour filer la métaphore, nous pouvons dire que le dictionnaire est la caisse à outils du traiteur de signal : c'est grâce aux outils contenus dedans (les atomes) qu'il pourra analyser les signaux.

Enfin, nous noterons que la définition de dictionnaire est plus souple que celle de trame (ou *frame* en anglais). Si le dictionnaire est seulement une collection redondante d'atomes, la trame est nécessairement de rang plein, *i.e.* $\text{rang}(\Phi) = N$ [Mal09].

1.1.3 Cas particulier de redondance : l'invariance par translation

La première forme de redondance que nous étudierons dans cette thèse découle du modèle invariant par translation, très utilisé pour l'analyse de signaux temps-séries.

Les signaux temps-séries sont composés d'une suite de valeurs représentant l'évolution temporelle d'une ou plusieurs quantités, comme les signaux audio [SL06, BD06], vidéo [Ols01] et audio-visuels [MJV+07, MVS09], les signaux électroencéphalographiques [JVLG05], les signaux électrocardiographiques [MGB+09], etc. Dans ce cadre, nous cherchons à ce que l'analyse effectuée soit invariante à la translation temporelle, en anglais *shift-invariance* (SI), *time-invariance* ou *translation-invariance*. Nous voulons décomposer avec parcimonie le signal temporel y comme la somme de quelques structures courtes, appelées noyaux, qui sont caractérisées indépendamment de leurs positions temporelles. Ces noyaux forment un dictionnaire compact Ψ et, de plus, ce modèle évite les effets de bloc pour l'analyse de signaux de large période [LS99, SL05a]. Dans [Blu05, Chap. 3], Blumensath distingue deux approches :

- le cas invariant par translation (*shift-invariant*) : si $y(t) \mapsto x(t)$, alors $y(t + t_0) \mapsto x(t + t_0)$,
- le cas consistant par translation (*shift-consistent*) : si $y(t) \mapsto x(t)$, alors $y(t + t_0) \mapsto x(t)$,

avec $y(t)$ et $y(t + t_0)$ ne représentant pas des échantillons, mais le signal y et sa version translatée de t_0 échantillons. On parle aussi de décalage ou de retard.

$$\Psi = \left\{ \psi_1 = \begin{bmatrix} 0 \\ \clubsuit_1 \\ \clubsuit_2 \\ \clubsuit_3 \\ 0 \end{bmatrix} \quad , \quad \psi_2 = \begin{bmatrix} 0 \\ \diamondsuit_1 \\ \diamondsuit_2 \\ \diamondsuit_3 \\ \diamondsuit_4 \\ \diamondsuit_5 \\ 0 \end{bmatrix} \quad , \quad \psi_3 = \begin{bmatrix} 0 \\ \spadesuit_1 \\ \spadesuit_2 \\ \spadesuit_3 \\ \spadesuit_4 \\ 0 \end{bmatrix} \right\}$$

Fig. 1.3 – Exemple d'un dictionnaire de noyaux Ψ, avec $L = 3$.

$$\Phi = \left[\begin{array}{ccccc|ccccc|ccccc}
0 & 0 & 0 & & 0 & 0 & 0 & & 0 & 0 & 0 & 0 & & 0 & 0 \\
\clubsuit_1 & 0 & 0 & & 0 & \diamondsuit_1 & 0 & & 0 & \spadesuit_1 & 0 & 0 & & 0 & 0 \\
\clubsuit_2 & \clubsuit_1 & 0 & & 0 & \diamondsuit_2 & 0 & & 0 & \spadesuit_2 & 0 & 0 & & 0 & 0 \\
\clubsuit_3 & \clubsuit_2 & \clubsuit_1 & & 0 & \diamondsuit_3 & 0 & & 0 & \spadesuit_3 & 0 & 0 & & 0 & 0 \\
0 & \clubsuit_3 & \clubsuit_2 & & 0 & \diamondsuit_4 & 0 & & 0 & \spadesuit_4 & 0 & 0 & & 0 & 0 \\
0 & 0 & \clubsuit_3 & & 0 & \diamondsuit_5 & 0 & & 0 & 0 & 0 & 0 & & 0 & 0 \\
0 & 0 & 0 & & 0 & 0 & 0 & & 0 & 0 & 0 & 0 & & 0 & 0 \\
 & & & \ddots & & & & \ddots & & & & & \ddots & & \\
0 & 0 & 0 & & 0 & 0 & \diamondsuit_1 & & 0 & 0 & \spadesuit_1 & & & 0 & \\
0 & 0 & 0 & & \clubsuit_1 & 0 & \diamondsuit_2 & & 0 & 0 & \spadesuit_2 & & & 0 & \\
0 & 0 & 0 & & \clubsuit_2 & 0 & \diamondsuit_3 & & 0 & 0 & \spadesuit_3 & & & 0 & \\
0 & 0 & 0 & & \clubsuit_3 & 0 & \diamondsuit_4 & & 0 & 0 & \spadesuit_4 & & & 0 & \\
0 & 0 & 0 & & 0 & 0 & \diamondsuit_5 & & 0 & 0 & 0 & & & 0 & \\
 & & & \ddots & & & & \ddots & & & & & \ddots & & \\
0 & 0 & 0 & & 0 & 0 & 0 & & \diamondsuit_2 & 0 & 0 & & & \spadesuit_1 & \\
0 & 0 & 0 & & 0 & 0 & 0 & & \diamondsuit_3 & 0 & 0 & & & \spadesuit_2 & \\
0 & 0 & 0 & & 0 & 0 & 0 & & \diamondsuit_4 & 0 & 0 & & & \spadesuit_3 & \\
0 & 0 & 0 & & 0 & 0 & 0 & & \diamondsuit_5 & 0 & 0 & & & \spadesuit_4 & \\
0 & 0 & 0 & & 0 & 0 & 0 & & 0 & 0 & 0 & & & 0 &
\end{array}\right]$$

Fig. 1.4 – Exemple du dictionnaire d'atomes $\Phi \in \mathbb{R}^{N \times M}$ généré, grâce au modèle d'invariance par translation, à partir du dictionnaire de noyaux Ψ représenté en Fig. 1.3.

Les $L \ll M$ noyaux translatables (aussi appelés fonctions génératrices ou mères) du dictionnaire Ψ sont répliqués à tous les échantillons pour fournir les M atomes (appelés fonctions de base) du dictionnaire Φ. Les N échantillons du signal y, le résidu ϵ, et les atomes ϕ_m sont indexés par la variable t. Les noyaux

$\{\psi_l\}_{l=1}^L$ peuvent avoir des longueurs temporelles différentes notées $\{T_l\}_{l=1}^L$. Le noyau $\psi_l(t)$ est translaté à l'échantillon τ pour générer l'atome $\psi_l(t-\tau)$: une complétion de zéros est effectuée pour qu'il ait N échantillons. Le sous-ensemble $\sigma_l \subset \mathbb{N}$ collecte toutes les translations actives τ du noyau $\psi_l(t)$. Pour les quelques noyaux qui génèrent tous les atomes, l'Eq. (1.2) devient :

$$y(t) = \sum_{m=1}^{M} x_m \phi_m(t) + \epsilon(t) \tag{1.3}$$

$$= \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau} \psi_l(t-\tau) + \epsilon(t)\,. \tag{1.4}$$

Ainsi, le signal y est vu comme la somme de quelques noyaux translatables ψ_l. Cette représentation est à la fois invariante par translation temporelle, mais aussi invariante à l'échelle grâce aux coefficients.

En raison de ce modèle invariant par translation, le dictionnaire d'atomes Φ est dit convolutif. Il est désormais la concaténation de L matrices de Toeplitz [PABD06, BD06], et il est L fois surcomplet/redondant. En résumé, avec ce modèle de représentation invariante par translation temporelle, chaque noyau est potentiellement démultiplié en une famille d'atomes translatés à tous les échantillons. Ainsi, le dictionnaire de noyaux invariants génère un dictionnaire d'atomes très redondant, qui est donc idéal pour représenter les données étudiées redondantes.

Ce modèle est illustré en Fig. 1.3 et 1.4. Un dictionnaire Ψ de $L = 3$ noyaux est affiché en Fig. 1.3. Les noyaux sont composés des symboles $\clubsuit_.$, $\diamondsuit_.$ et $\spadesuit_.$ qui représentent des valeurs réelles. Dans cet exemple, la longueur du noyau ψ_1 est $T_1 = 5$, celle du noyau ψ_2 est $T_2 = 7$ et celle du noyau ψ_3 est $T_3 = 6$. Comme illustré en Fig. 1.4, le dictionnaire d'atomes $\Phi \in \mathbb{R}^{N \times M}$ est généré, grâce au modèle d'invariance par translation, à partir du dictionnaire de noyaux translatables Ψ. De plus, aux effets de bords près, nous avons $M \approx L \times N$. Remarquons qu'il est préférable d'avoir des noyaux à bords nuls pour éviter les discontinuités dans les décompositions réalisées avec ce dictionnaire.

1.2 Approximation parcimonieuse

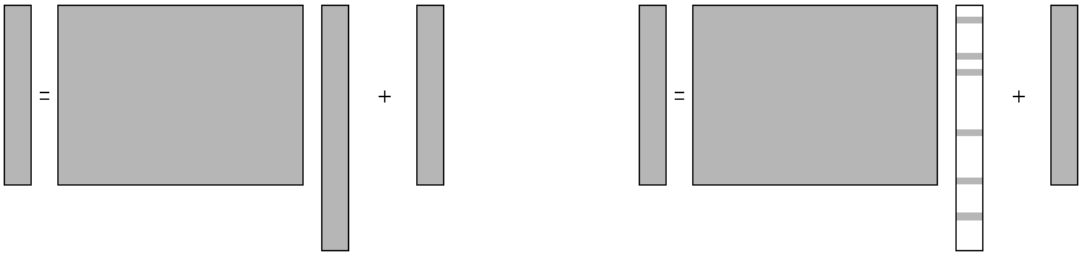
Nous nous intéressons à l'estimation des coefficients x pour un dictionnaire Φ donné. Comme le dictionnaire est redondant ($M > N$), le système linéaire (1.1) est sous-déterminé et possède de multiples solutions. L'introduction de contraintes sur le vecteur x telles que la positivité, la parcimonie, etc. permet de régulariser la solution. Dans cette thèse, nous nous intéresserons à la contrainte de parcimonie qui est la clé de l'Analyse en Composantes Parcimonieuses introduite en Section 1.1.2.

L'approximation parcimonieuse (*sparse approximation* en anglais) est donc l'approche transformant une décomposition linéaire basique, comme illustrée en Fig. 1.5(a), en une décomposition linéaire parcimonieuse, comme illustrée en Fig. 1.5(b), où seuls quelques coefficients sont non nuls.

1.2.1 Formulations de l'approximation parcimonieuse

La pseudo-norme ℓ_0 est utilisée pour formaliser la parcimonie du vecteur de coefficients x : $\|x\|_0$ est défini comme le cardinal du support de x, *i.e.* le nombre d'éléments non nuls de x. La décomposition du signal y sous contrainte de parcimonie s'écrit :

$$\min_x \|x\|_0 \;\text{ t.q. }\; \| y - \Phi x \|_2^2 \leq \xi_0\,, \tag{1.5}$$



(a) $y = \Phi x + \epsilon$.

(b) $y = \Phi x + \epsilon$ avec x parcimonieux.

Fig. 1.5 – Illustration des décompositions linéaires, basique (a) et avec contrainte de parcimonie (b).

avec ξ_0 une constante. La formulation (1.5) possède :

- un terme de minimisation pour obtenir le vecteur x le plus parcimonieux,
- et un terme quadratique d'attache aux données.

En général, trouver la solution la plus parcimonieuse au problème (1.5) est NP-difficile[1] [Dav94]. Ce problème peut être reformulé en une version contrainte des moindres carrés :

$$\min_x \ \| y - \Phi x \|_2^2 \ \text{t.q.} \ \|x\|_0 \leq K \,, \tag{1.6}$$

avec K la constante de parcimonie. Les algorithmes de poursuite ℓ_0 (cf. Section 1.2.4) résolvent le problème (1.6) séquentiellement en incrémentant la valeur K itérativement. Cependant, cette optimisation est non-convexe : la solution obtenue peut correspondre à un minimum local. Parmi les multiples poursuites ℓ_0, nous mentionnons le *Matching Pursuit* (MP) proposé par Mallat et Zhang [MZ93] et l'*Orthogonal Matching Pursuit* (OMP) [PRK93] qui seront étudiés plus particulièrement dans cette thèse. Leurs solutions sont sous-optimales parce que l'identification du support n'est pas garantie, spécialement pour une forte cohérence du dictionnaire μ_Φ (cf. Section 1.4.1). Cependant, elles sont rapides pour la recherche de très peu de coefficients [Tro04].

Une autre approche consiste à relaxer le terme de parcimonie de pseudo-norme ℓ_0 en une norme ℓ_1. Le problème résultant est appelé *Basis Pursuit Denoising* [CDS98] :

$$\min_x \ \|x\|_1 \ \text{t.q.} \ \| y - \Phi x \|_2^2 \leq \xi_1 \,, \tag{1.7}$$

avec ξ_1 une constante. Ce problème est une optimisation convexe avec un unique minimum, ce qui est un avantage par rapport aux algorithmes de poursuites ℓ_0. Sous certaines conditions strictes [BDE09], la solution peut être équivalente au problème ℓ_0 (cf. Section 1.4.1) et elle est optimale. Le problème (1.7) peut être reformulé par le *Least Absolute Shrinkage and Selection Operator* (LASSO) [Tib96] :

$$\min_x \ \| y - \Phi x \|_2^2 \ \text{t.q.} \ \|x\|_1 \leq K_1 \,, \tag{1.8}$$

ou encore à l'aide du Lagrangien :

$$\min_x \ \| y - \Phi x \|_2^2 + \lambda \|x\|_1 \ . \tag{1.9}$$

Ces trois formulations sont équivalentes pour des valeurs de paramètres bien choisies. Différents algorithmes pour résoudre ces problèmes sont revus en Section 1.2.4. Cependant, une forte cohérence μ_Φ ne peut pas assurer que ces algorithmes identifieront le support optimal x [BDE09] et, quand c'est le cas, la convergence peut être très lente.

1. En anglais, *NP-hard* pour *non-deterministic polynomial-time hard*.

1.2.2 Présentation MP et OMP

Dans ce paragraphe, l'OMP [PRK93] est expliqué pas à pas, et le MP [MZ93] est montré comme une restriction sous-optimale de l'OMP. Etant donné un dictionnaire redondant Φ, l'OMP produit une approximation parcimonieuse du signal y, comme décrit par l'Algorithme 1. Il résout le problème des moindres carrés (1.6) sur un sous-espace poursuivi par une sélection itérative des atomes. Cet algorithme, présenté ici dans le cas réel, a été introduit directement en complexe par Pati *et al.* [PRK93].

Après l'initialisation du résidu courant sur le signal d'entrée (étape 1), à l'itération courante k, l'OMP sélectionne l'atome qui produit la plus forte décroissance (en valeur absolue) de l'erreur quadratique moyenne $\left\|\epsilon^{k-1}\right\|_2^2$. En notant $\epsilon^{k-1} = x_m \phi_m + \epsilon^k$, nous avons :

$$\frac{\partial \left\| \epsilon^{k-1} \right\|_2^2}{\partial x_m} = 2\, {\phi_m}^T\, \epsilon^{k-1} = 2 \left\langle \epsilon^{k-1}, \phi_m \right\rangle , \tag{1.10}$$

avec l'erreur ϵ^k orthogonale aux atomes sélectionnés. Ainsi, ce critère est équivalent à trouver l'atome le plus corrélé au résidu. Pour cela, le produit scalaire entre le résidu ϵ^{k-1} et chaque atome ϕ_m est calculé (étape 4). Ensuite, le maximum des valeurs absolues est recherché (étape 6) pour sélectionner l'atome optimal de l'itération courante, noté ϕ_{m^k}.

Un dictionnaire actif $D^k \in \mathbb{R}^{N \times k}$ est formé, collectant les k atomes sélectionnés (étape 7). Les coefficients x^k sont calculés via la projection orthogonale de y sur le sous-espace sélectionné D^k (étape 8) :

$$x^k = \arg\min_x \left\| y - D^k x \right\|_2^2 . \tag{1.11}$$

Ce calcul est souvent effectué récursivement par différentes méthodes, utilisant la valeur du produit scalaire courant $C^k_{m^k}$: par la factorisation QR [DMZ94], par la factorisation de Cholesky [CARKD99], ou par l'inversion matricielle par bloc [PRK93]. Le vecteur de coefficients obtenu $x^k = [x_{m^1};\, x_{m^2} \ldots x_{m^k}]$ est donc réduit à ces coefficients actifs (*i.e.* non nuls). Nous noterons [;] la concaténation verticale, et [,] la concaténation horizontale.

Algorithm 1 : $x = \text{OMP}\ (y, \Phi)$

1: **initialisation** : $k = 1,\ \epsilon^0 = y,$ dictionnaire $D^0 = \varnothing$
2: **repeat**
3: **for** $m \leftarrow 1, M$ **do**
4: Produits scalaires : $C^k_m \leftarrow \left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle$
5: **end for**
6: Sélection : $m^k \leftarrow \arg\max_m \left| C^k_m \right|$
7: Dictionnaire actif : $D^k \leftarrow \left[\, D^{k-1}, \phi_{m^k} \,\right]$
8: Coefficients actifs : $x^k \leftarrow \arg\min_x \left\| y - D^k x \right\|_2^2$
9: Résidu : $\epsilon^k \leftarrow y - D^k x^k$
10: $k \leftarrow k + 1$
11: **until** critère d'arrêt

Différents critères d'arrêt (étape 11) peuvent être utilisés : un seuillage sur la valeur k de l'itération en cours, un seuillage sur la rMSE $\left\|\epsilon^k\right\|_2^2 / \left\|y\right\|_2^2$, ou un seuillage sur la décroissance de la rMSE. A la fin,

l'OMP fournit une approximation K-parcimonieuse du signal y :

$$\hat{y}^K = \sum_{k=1}^{K} x_{m^k}\, \phi_{m^k} \,. \tag{1.12}$$

La convergence de l'OMP est démontrée dans [PRK93], et ses propriétés d'identification de support sont analysées dans [DVT96, Tro04]. Pour des applications temps-réel, l'OMP est un excellent compromis entre temps de calcul et performances [Tro04].

La différence essentielle entre le MP et l'OMP est le calcul des coefficients : dans le MP décrit dans l'Algorithme 2, ils sont calculés comme étant la valeur de la corrélation optimale $C^k_{m^k}$, *i.e.* celle dont la valeur absolue est maximale. Cette façon plus rapide est cependant sous-optimale quand les atomes ne sont pas orthogonaux (incluant les cas de recouvrements d'atomes du dictionnaire actif, *i.e.* quand leurs supports sont communs). Dans ces cas, la solution x obtenue n'est pas celle des moindres carrés. Cela a aussi pour conséquence de modifier la sélection des atomes actifs lors des itérations internes.

Algorithm 2 : $x = \mathrm{MP}\ (y, \Phi)$

initialisation : $k = 1,\ \epsilon^0 = y,$ dictionnaire $D^0 = \varnothing$
repeat
  for $m \leftarrow 1, M$ **do**
    Produits scalaires : $C^k_m \leftarrow \left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle$
  end for
  Sélection : $m^k \leftarrow \arg\max_m \left|\, C^k_m \,\right|$
  Dictionnaire actif : $D^k \leftarrow \left[\, D^{k-1}, \phi_{m^k} \,\right]$
  Coefficients actifs : $x^k \leftarrow \left[\, x^{k-1}; C^k_{m^k} \,\right]$
  Résidu : $\epsilon^k \leftarrow \epsilon^{k-1} - C^k_{m^k} \phi_{m^k}$
  $k \leftarrow k + 1$
until critère d'arrêt

1.2.3 OMP dans le cas d'invariance par translation

Pour décomposer des signaux temporels, nous étudions le cas d'invariance par translation et nous détaillons les étapes de l'OMP qui changent par rapport au cas classique décrit dans l'Algorithme 1. En ré-écrivant le problème étudié dans le formalisme invariant par translation, nous avons :

$$\min_x \left\| y(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau} \psi_l(t-\tau) \right\|_2^2 \text{ t.q. } \|x\|_0 \leq K \,. \tag{1.13}$$

Dans l'étape 4 de l'Algorithme 3, le produit scalaire entre le résidu et chaque atome ϕ_m est maintenant remplacé par la corrélation entre le résidu et chaque noyau ψ_l, généralement calculée par transformée de Fourier rapide. La corrélation non-circulaire $\Gamma \in \mathbb{R}^{N_1+N_2-1}$ entre les signaux $y_1(t) \in \mathbb{R}^{N_1}$ et $y_2(t) \in \mathbb{R}^{N_2}$ est définie par :

$$\Gamma\left\{y_1, y_2\right\}(\tau) = \langle y_1(t), y_2(t-\tau) \rangle = {y_2}^T(t-\tau)\, y_1(t) \,. \tag{1.14}$$

L'étape 6 sélectionne l'atome optimal, caractérisé par son indice de noyau l^k et sa position temporelle τ^k. Les coefficients $x^k = \left[x_{l^1,\tau^1}; x_{l^2,\tau^2} \ldots x_{l^k,\tau^k}\right]$ sont obtenus par projection orthogonale de y sur le dictionnaire actif D^k. A la fin, l'approximation K-parcimonieuse (1.12) devient :

$$\hat{y}^K(t) = \sum_{k=1}^{K} x_{l^k,\tau^k}\, \psi_{l^k}(t-\tau^k)\,. \tag{1.15}$$

Algorithm 3 : $x = \text{OMP}\ (y, \Psi)$

initialisation : $k = 1,\ \epsilon^0 = y,$ dictionnaire $D^0 = \varnothing$
repeat
  for $l \leftarrow 1, L$ **do**
    Corrélations : $C_l^k(\tau) \leftarrow \Gamma\left\{\ \epsilon^{k-1},\ \psi_l\ \right\}(\tau)$
  end for
  Sélection : $(l^k, \tau^k) \leftarrow \arg\max_{l,\tau}\ \left|\ C_l^k(\tau)\ \right|$
  Dictionnaire actif : $D^k \leftarrow \left[\ D^{k-1}, \psi_{l^k}(t-\tau^k)\ \right]$
  Coefficients actifs : $x^k \leftarrow \arg\min_x \left\|\ y - D^k x\ \right\|_2^2$
  Résidu : $\epsilon^k \leftarrow y - D^k x^k$
  $k \leftarrow k+1$
until critère d'arrêt

Dans le cas de l'invariance par translation, les coefficients $\left\{x_{l^k,\tau^k}\right\}_{k=1}^{K}$ de la décomposition K-parcimonieuse sont présentés à l'aide d'un spikegramme [2] [SL05a]. Il s'agit d'une représentation temps-noyaux, généralisant la représentation temps-fréquence donnée par un spectrogramme (où les noyaux sont des atomes de Fourier). Pour chacun des K atomes actifs sélectionnés, le spikegramme condense trois informations :

- la position temporelle τ^k en abscisse,
- l'indice de noyaux l^k en ordonnée,
- l'amplitude du coefficient x_{l^k,τ^k} en échelle de couleur.

Ce type de représentation fournit une excellente visualisation de la décomposition parcimonieuse du signal étudié.

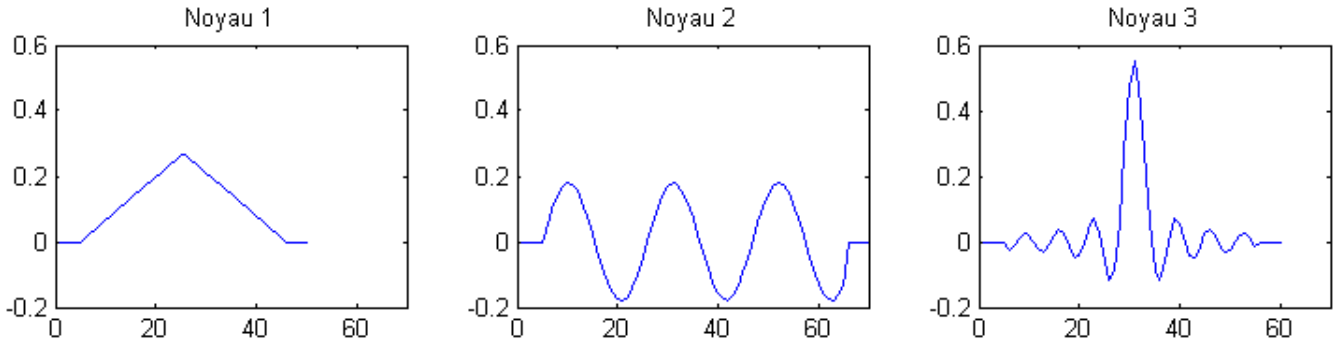


Fig. 1.6 – Exemple d'un dictionnaire Ψ composé de $L = 3$ noyaux.

2. Version francisée de *spikegram* signifiant diagramme d'impulsions.

Pour illustrer cela, nous fabriquons un dictionnaire de synthèse Ψ composé de $L = 3$ noyaux et représenté en Fig. 1.6. Si les signaux sont définis verticalement dans les équations, nous les affichons horizontalement sur les figures. La Fig. 1.7 illustre un exemple de spikegramme utilisé avec le dictionnaire de synthèse et $K = 5$ atomes sélectionnés sur un signal de $N = 256$ échantillons. Les atomes actifs sont définis aux échantillons $\sigma_1 = \{10; 150\}$, $\sigma_2 = \{180\}$ et $\sigma_3 = \{70; 100\}$. Le signal original y est représenté en (a), les contributions du noyau ψ_1 en (b) avec les coefficients $x_{1,10} = 3$ et $x_{1,150} = -3$, la contribution du noyau ψ_2 en (c) avec le $x_{2,180} = 1$, les contributions du noyau ψ_3 en (d) avec les coefficients $x_{3,70} = 1$ et $x_{3,100} = 1$, l'approximation $\hat{y}^K$ et le résidu ϵ en (e). Tout en bas, le spikegramme affiche l'amplitude de chaque coefficient en fonction de la position temporelle et de l'indice de noyaux. Remarquons que le *spike* est situé au début du noyau, alors que certaines représentations le situe au centre. Cet outil fournit une visualisation condensée des contributions des différents noyaux pour l'approximation parcimonieuse du signal étudié.

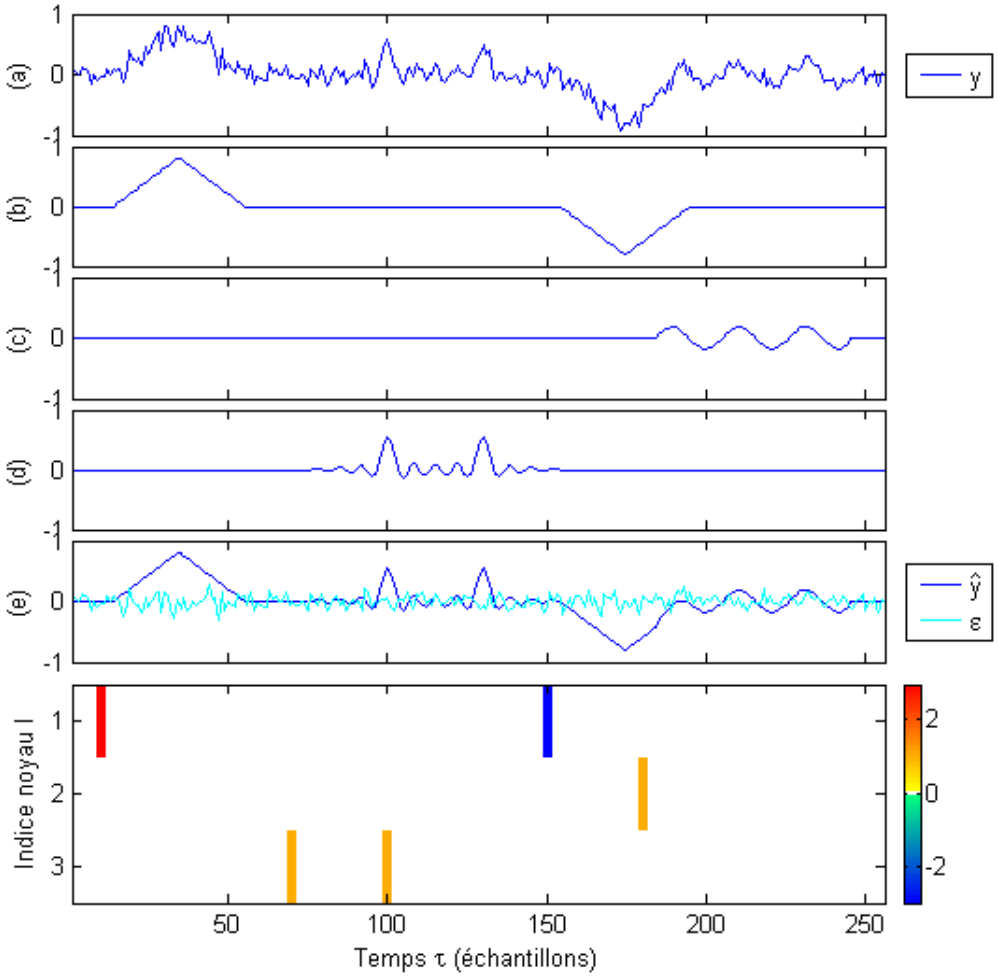


Fig. 1.7 – Illustration de la représentation temps-noyaux sur un exemple avec le dictionnaire de la Fig. 1.6 et avec $K = 5$ atomes actifs sélectionnés sur un signal de $N = 256$ échantillons. Le signal original y est représenté en (a), les contributions du noyau ψ_1 en (b) avec les coefficients $x_{1,10} = 3$ et $x_{1,150} = -3$, la contribution du noyau ψ_2 en (c) avec le $x_{2,180} = 1$, les contributions du noyau ψ_3 en (d) avec les coefficients $x_{3,70} = 1$ et $x_{3,100} = 1$, l'approximation $\hat{y}^K$ et le résidu ϵ en (e). Tout en bas, le spikegramme : l'amplitude (échelle de couleur) de chaque coefficient x_{l^k,τ^k} est affichée en fonction de la position temporelle τ^k (abscisse) et de l'indice de noyaux l^k (ordonnée).

1.2.4 Autres algorithmes d'approximation parcimonieuse

Il existe de multiples méthodes d'approximation parcimonieuse, avec de nombreuses reprises et améliorations : les articles de Bruckstein *et al* [BDE09] et de Tropp et Wright [TW10] en font une excellente revue. Nous ne citerons ici que les principales méthodes.

Méthodes ℓ_0

De nombreuses améliorations et extensions de l'OMP ont eu lieu pour résoudre le problème (1.6). Ces méthodes sont dites gloutonnes, car elles recherchent des minima successifs en espérant atteindre le minimum global. Généralement très rapides, elles ont le risque d'être bloquées dans un minimum local.

L'algorithme *Orthogonal Least-Squares* (OLS) [CBL89, CW95] est plus performant que l'OMP car il sélectionne l'atome qui produit la plus forte décroissance de $\left\|\epsilon^k\right\|_2^2$ et non de $\left\|\epsilon^{k-1}\right\|_2^2$ (cf. Section 1.2.2). Il a été redécouvert sous de multiples noms tels que *Order Recursive* MP (ORMP) [CARKD99], *Optimized* OMP (OOMP) [RNL02], *pre-fitting* OMP [VB02] et l'article [BD07] lève les multiples confusions entre ces deux algorithmes. Notons que l'algorithme *Single Best Replacement* (SBR) [SIBD11] propose une extension *forward-backward* de l'OLS, *i.e.* que les atomes peuvent être ajoutés (étape avant ou *forward*) mais aussi retirés (étape arrière ou *backward*).

Une nouvelle génération d'algorithmes effectue une sélection, non plus itérative, mais simultanée des K atomes. Le *Stagewise* OMP (StOMP) [DTDS06] sélectionne plusieurs atomes à chaque itération grâce à un seuillage en valeur des corrélations (la valeur du seuil dépendant de la norme du résidu). Le *Regularized* OMP (ROMP) [NV09], proche du StOMP, sélectionne plusieurs atomes puis en rejette certains de telle sorte que les coefficients gardés ne diffèrent pas plus que d'un facteur 2. Le *Subspace Pursuit* [DM09] ou le CoSaMP [NT09] réévaluent à chaque itération les K atomes candidats en rejettant ceux qui ne sont pas optimaux : le sous-espace poursuivi est donc remis intégralement en doute à chaque itération. Cette approche est améliorée par le *Two Stage Thresholding* [MD10] qui s'affranchit notamment du choix de l'hyperparamètre K.

D'autres approches ℓ_0 existent comme le *Iterative Hard Thresholding* (IHT) [BD09, BD10] qui effectue un seuillage dur itératif (*i.e.* basé sur le cardinal) pour sélectionner simultanément les K atomes optimaux. Il est très rapide car il ne possède pas d'étape de projection orthogonale. Le *Gradient Pursuit* [BD08] effectue la même sélection que l'OMP, mais la solution des moindres carrés est calculée de façon approchée par descente de gradient (directions conjuguées). Cet algorithme accélère le temps de calcul mais les performances sont dégradées. Enfin, la méthode ℓ_0 lissée (SL0) [MBZJ09] approche la contrainte ℓ_0 par une fonction continue paramétrée. Au cours des itérations, le paramètre évolue vers 0, ce qui fait tendre la fonction vers un pic qui produit l'approximation ℓ_0.

Méthodes ℓ_1

Le *Basis Pursuit Denoising* [CDS98] autorise l'utilisation de la programmation linéaire pour résoudre le problème (1.9) via les algorithmes du Simplex ou de point intérieur. Différentes méthodes à base de point intérieur sont employées, utilisant les équations primal-dual [CDS98, Sau02], l'optimisation quadratique [KKL+07] ou encore la reformulation du problème (1.7) en programmation sur un cône de second ordre [CR05]. Ces méthodes sont réputées être robustes mais lentes.

L'homotopie [OPT00, MCW05] résout le problème (1.9) : en utilisant la transformation homotopique de la norme ℓ_2 vers la norme ℓ_1 selon le paramètre λ, elle fait varier ce paramètre λ de l'infini à 0 en construisant un chemin linéaire par morceau. On entend par chemin de régularisation l'ensemble des solutions x d'un problème pénalisé en fonction de la valeur du paramètre de régularisation, soit $x(\lambda)$. Alors que l'homotopie est une méthode *forward-backward* aussi appelée LARS, le *Least Angle Regression* (LAR) [EHJT04] est seulement une version *forward* qui recrute les atomes de sorte qu'ils soient équicorrélés au résidu. Les liens entre l'homotopie, le LAR et l'OMP sont discutés dans [DT08].

Les méthodes à base de seuillage doux du gradient de 1$^{\text{er}}$ ordre résolvent itérativement le problème (1.9). Cette approche est appelée *Iterative Shrinkage Tresholding* (IST) [DDDM04] ou *splitting operator* [CW05]. Il existe de nombreuses améliorations comme le *Two-Step* IST (TwIST) [BDF07] qui prend en compte les valeurs des deux dernières itérations, le *Gradient Projection for Sparse Reconstruction* (GPSR) [FNW07] qui applique un gradient projeté sur une formulation quadratique du problème (1.9) obtenue en séparant la norme ℓ_1 en deux parties (positive et négative), le *Fixed-Point Continuation* (FPC) [HYZ08] qui accélère l'IST en évitant les démarrages à froid du paramètre de régularisation, et le *Sparse Reconstruction by Separable Approximation* (SpaRSA) [WNF09] qui généralise et accélère l'IST notamment grâce à une variante de Barzilai-Borwein pour le choix du pas de descente de gradient. D'autres approches récentes optimisent la vitesse de convergence à l'aide du gradient de Nesterov [BT09, BBC11].

Certaines méthodes introduisent des coefficients de pondération, comme le *Iterative Reweighted Least-Squares* (IRLS) [YBW85] connu aussi sous le nom de FOCUSS [GR97, RKD99] ou encore le *Iterative Reweighted $\ell 1$-Minimization* (IRℓ1) [CWB08]. D'autres approches existent comme le *Block Coordinate Relaxation* (BCR) [SBT00] qui effectue une IST sur des sous-dictionnaires, le *Parallel Coordinate Descent* [Ela06] qui parallélise les mises à jour individuelles scalaires des coefficients, ou encore la méthode des directions alternées (ADM) [YZ11, NWY11] qui effectue une optimisation alternée des variables primales et duales du Lagrangien étendu.

Autres méthodes

Il existe d'autres formes de régularisation très utilisées en traitement du signal et en statistique. La méthode *elastic-net* [Zou05] est un compromis entre une pénalité ℓ_1 et une pénalité ℓ_2 (aussi appelée régularisation de Tikhonov ou *ridge regression*). La régularisation de Variation Totale (TV) [ROF92] met une pénalité ℓ_1 sur le gradient, ce qui engendre une parcimonie des variations, utile pour le débruitage d'images.

Pour conclure cette section, nous avons présenté les principales méthodes d'approximation parcimonieuse. Nous avons détaillé l'OMP dans le cas de l'invariance par translation, car cette méthode sera utilisée et étendue dans la suite.

1.3 Apprentissage de dictionnaire

Dans la section précédente, nous avons étudié comment estimer avec parcimonie les coefficients de décomposition en considérant le dictionnaire fixé. Mais le contenu du dictionnaire conditionne cette estimation. Dans cette section, nous nous intéressons à l'estimation du dictionnaire optimal favorisant

au mieux les approximations parcimonieuses. Dans un premier temps, nous expliquerons le principe de l'apprentissage de dictionnaire, puis nous ferons l'état de l'art des méthodes.

1.3.1 Principe de l'apprentissage de dictionnaire

Dans ce paragraphe, nous expliquons le principe de l'apprentissage de dictionnaire, et nous formalisons le problème.

A la recherche du dictionnaire idéal

Afin d'obtenir la solution la plus parcimonieuse au problème (1.6), il faut que le dictionnaire soit adapté au signal. Quand nous traitons un ensemble de signaux, il faut que le dictionnaire soit conjointement adapté à tous ces signaux. Imaginons un dictionnaire Φ contenant tous les motifs possibles : cela permettrait à n'importe quel signal d'être décomposé avec parcimonie. Cependant, ce dictionnaire serait trop grand à stocker, et l'estimation des coefficients serait trop lourde. Un choix doit donc être fait sur le dictionnaire utilisé et, pour cela, trois approches sont possibles.

Premièrement, nous pouvons choisir parmi les dictionnaires génériques [JDCP11] tels que les atomes de Fourier, les ondelettes [Mal09], les ridgelettes [CD99], les curvelettes [CD00], etc. Ces dictionnaires génériques ont l'avantage d'avoir des atomes analytiques et paramétrés, mais leurs morphologies influencent intrinsèquement l'analyse. Si nous utilisons des ondelettes, nous trouverons des ondelettes dans les signaux, si nous utilisons des curvelettes, nous trouverons des curvelettes, etc. Un expert doit donc déterminer quel dictionnaire employer en fonction des signaux étudiés : les ondelettes sont adaptées pour représenter les textures, les curvelettes pour les discontinuités, etc. Ainsi, pour choisir convenablement un dictionnaire qui soit pertinent, nous devons avoir des *a priori* sur les motifs attendus dans les signaux, ce qui peut constituer un inconvénient de taille. Cette première approche est la plus souvent utilisée en traitement du signal, et Rubinstein *et al.* retracent bien l'évolution du choix des dictionnaires [RBE10].

Deuxièmement, plusieurs de ces dictionnaires génériques peuvent être concaténés, permettant ainsi aux différentes composantes du signal d'être séparées, chacune étant parcimonieuse sur le sous-dictionnaire qui lui est dédié [CDS98, SED04, FSBM10]. Si cette approche est plus flexible, il est toujours nécessaire d'avoir des *a priori* sur les motifs attendus pour choisir chacun des sous-dictionnaires.

Troisièmement, nous utilisons les données pour choisir le dictionnaire optimal (en anglais, ce type d'approche est qualifié de *data-driven*). L'apprentissage de dictionnaire consiste à inférer un dictionnaire non-paramétrique à partir d'un ensemble de données de telle sorte qu'il soit adapté à ces données, *i.e.* qu'il fournisse conjointement des approximations parcimonieuses pour tous les signaux de cet ensemble [TF11]. L'apprentissage de dictionnaire est aussi appelé encodage parcimonieux, car les décompositions effectuées *a posteriori* sur ce dictionnaire appris sont parcimonieuses. En anglais, on parle de *sparse coding*, terme très souvent confondu avec *sparse approximation*.

Formulation de l'apprentissage de dictionnaire

Nous disposons d'un ensemble de signaux d'entraînement $Y = [\, y_p \,]_{p=1}^{P} \in \mathbb{R}^{N \times P}$ qui sont représentatifs de l'ensemble étudié. Nous définissons aussi $X = [\, x_p \,]_{p=1}^{P} \in \mathbb{R}^{M \times P}$ qui contient les vecteurs de coefficients correspondants. Le but est d'apprendre le dictionnaire le plus adapté à l'ensemble des signaux étudiés Y, *i.e.* que les signaux de Y ont des approximations parcimonieuses sur ce dictionnaire.

La formulation générale de l'apprentissage de dictionnaire s'écrit :

$$\min_{\Phi,X} \| Y - \Phi X \|_F^2 \quad \text{t.q.} \quad \forall p \in \mathbb{N}_P, \|x_p\|_0 \leq K \quad \text{et} \quad \forall m \in \mathbb{N}_M, \|\phi_m\|_2 = 1 \,. \tag{1.16}$$

Le dictionnaire Φ est considéré normé. Cette contrainte sur la norme possède deux avantages :

- les coefficients x reflètent l'énergie de chaque atome dans le signal décomposé,
- cela évite l'indétermination du produit ΦX (qui est équivalent à un produit où X est multiplié par un facteur, et Φ divisé par ce même facteur).

En résumé, l'apprentissage de dictionnaire consiste à adapter les atomes à la distribution statistique des données.

Les algorithmes d'apprentissage de dictionnaire, en anglais *Dictionary Learning Algorithm* (DLA), résolvent ce problème par une optimisation alternée entre l'approximation parcimonieuse des signaux et la mise à jour du dictionnaire. Les motifs élémentaires et énergétiques des signaux sont sélectionnés dans l'étape d'approximation parcimonieuse, et ensuite appris lors de l'étape de mise à jour du dictionnaire. L'initialisation du dictionnaire peut se faire sur des atomes de bruit blancs, ou sur les signaux d'apprentissage.

Remarquons que ce problème d'apprentissage de dictionnaire peut être lui aussi relaxé en utilisant une norme ℓ_1 : si chacune des deux étapes est désormais convexe, le problème global reste non-convexe.

Remarques

Les atomes appris n'appartiennent pas à des dictionnaires classiques, et ne sont pas paramétrés à partir d'un atome prototype. Ils sont appropriés aux données et à l'application considérée.

En neurosciences, Olshausen et Field [OF97] étudient des images naturelles et apprennent un dictionnaire dont les atomes ressemblent aux champs récepteurs du cortex visuel primaire (V1) des mammifères. En étudiant des signaux de paroles humaines, Smith et Lewicki [SL06] font émerger des atomes ressemblant aux filtres de la cochlée des mammifères. Ces constatations empiriques montrent que le système sensoriel des mammifères est bien adapté à leur environnement naturel, mais elles montrent surtout la puissance de l'apprentissage de dictionnaire. En considérant une base de données ayant acquis un phénomène, l'apprentissage de dictionnaire peut extraire les causes génératrices de ce phénomène. Les atomes appris correspondent aux causes sous-jacentes du phénomène observé et fournissent un encodage efficace. Cette approche est donc propice à la découverte de motifs informatifs, constituant une méthode d'exploration de données (*data mining* en anglais).

Les dictionnaires appris montrent de meilleures performances que les méthodes de l'état de l'art, comme pour le débruitage [EA06, MES08], le *demosaicing* et l'*inpainting* [MES08], la super-résolution [YWHM10, ZEP12], etc. Les domaines d'applications comprennent les images [OF97, KDMR+03, AEB06a, AE08], l'audio [LS99, SL05a, BD06], la video [Ols01], l'audio-visuel [MVS09] et l'électrocardiogramme [EAH00, ESH07].

1.3.2 Revue des méthodes d'apprentissage

Dans ce paragraphe, nous passons en revue les principaux algorithmes d'apprentissage de dictionnaire et leurs apports techniques à cette problématique.

Méthodes probabilistes

Les premières méthodes [OF96, OF97, LS98, LO99] se fondaient sur une approche probabiliste, en considérant la fonction de vraisemblance $\mathfrak{p}(Y|\Phi)$. Pour maximiser cette fonctionnelle, les P signaux sont d'abord considérés comme indépendants :

$$\mathfrak{p}(Y|\Phi) = \prod_{p=1}^{P} \mathfrak{p}(y_p|\Phi) \,. \tag{1.17}$$

D'autre part, pour chaque signal y_p :

$$\mathfrak{p}(y_p|\Phi) = \int \mathfrak{p}(y_p|x,\Phi) \cdot \mathfrak{p}(x)\, dx \,. \tag{1.18}$$

Le terme $\mathfrak{p}(y_p|x,\Phi)$ fournit le terme quadratique d'attache aux données sous hypothèse d'un bruit blanc Gaussien, et le terme $\mathfrak{p}(x)$ est utilisé pour introduire sur les coefficients un *a priori* de parcimonie de type Cauchy [OF97] ou Laplace [OF96, LO99]. Dans ce dernier cas, la formulation finale revient à celle donnée dans l'Eq. (1.9).
Pour résoudre ce problème, ces méthodes alternent entre deux étapes :

1. Φ est fixé et x est calculé pour chaque signal, par descente de gradient dans les premières versions [OF96, OF97], puis par approximation parcimonieuse [LS98, LO99],
2. x est fixé et Φ est mis à jour par descente de gradient, puis normalisé.

Une autre méthode consiste à maximiser $\mathfrak{p}(\Phi|Y) \propto \mathfrak{p}(Y|\Phi)\mathfrak{p}(\Phi)$ en utilisant la règle de Bayes [KDMR+03]. Cette approche ajoute donc un *a priori* sur le dictionnaire Φ, en considérant deux normalisations. Soit le dictionnaire est normé au sens classique, *i.e.* chaque colonne est de norme ℓ_2 égale à 1, soit la matrice Φ est de norme de Frobenius égale à 1.

Méthode ILS-DLA

Introduite par Engan *et al.*, la méthode des moindres carrés itératifs [ESH07], *Iterative Least-Squares* DLA (ILS-DLA), s'est originellement développée sous le nom de méthode des directions optimales [EAH00], *Method of Optimal Directions* (MOD). Elle est similaire à l'approche probabiliste, sous hypothèse de bruit Gaussien. Concernant la mise à jour, l'approche probabiliste utilise une optimisation à base de gradient alors que cette méthode calcule la pseudo-inverse de X. Avec un formalisme d'optimisation, cette méthode effectue l'approximation parcimonieuse de tous les signaux :

$$\min_X \; \| Y - \Phi X \|_F^2 \quad \text{t.q.} \quad \forall p \in \mathbb{N}_P, \|x_p\|_0 \leq K \,, \tag{1.19}$$

puis la mise à jour du dictionnaire :

$$\min_\Phi \; \| Y - \Phi X \|_F^2 \quad \text{t.q.} \quad \forall m \in \mathbb{N}_M, \|\phi_m\|_2 = 1 \,. \tag{1.20}$$

Cette étape est résolue par les moindres carrés : $\Phi = YX^T \left(XX^T\right)^{-1}$, qui s'avère être lourde pour des données de tailles importantes. Cette mise à jour est dite groupée, soit *batch* en anglais, car tous les signaux sont d'abord décomposés avant d'être tous utilisés pour la mise à jour du dictionnaire.

Méthode K-SVD

Introduite par Aharon *et al.*, la méthode K-SVD [AEB06a] suit le même schéma de mise à jour groupée, mais effectue une mise à jour simultanée du dictionnaire et des coefficients. Dans cet algorithme :

1. Φ est fixé et x est obtenu par approximation parcimonieuse,
2. le support de x est gardé et ensuite, Φ et x sont mis à jour par SVD en utilisant la première composante.

L'avantage majeur de la K-SVD sur la méthode ILS-DLA est que les atomes mis à jour sont directement normés. Il n'y a donc pas d'étape de re-normalisation des atomes, ce qui assure une stricte décroissance de l'erreur. Cependant, cette technique de mise à jour reste coûteuse pour de grosses bases de données.

Algorithm 4 : Φ = Batch_DLA $([y_p]_{p=1}^P)$

initialisation
repeat
  for $p \leftarrow 1, P$ **do**
    Approximation parcimonieuse de y_p
  end for
  Mise à jour du dictionnaire Φ
until critère d'arrêt

Algorithm 5 : Φ = Online_DLA $([y_p]_{p=1}^P)$

initialisation
repeat
  for $p \leftarrow 1, P$ **do**
    Approximation parcimonieuse de y_p
    Mise à jour du dictionnaire Φ
  end for
until critère d'arrêt

Méthodes *online*

Contrairement aux méthodes utilisant une mise à jour groupée, les récents DLAs utilisent une mise à jour en ligne [MBPS10, SE10], soit en anglais *online*, *continuous* ou *recursive*. Les signaux sont traités un par un et non plus de façon groupée. La mise à jour a lieu après l'approximation parcimonieuse de chaque signal y_p : il y a donc P fois plus de mises à jour que précédemment. L'ordre de traitement des signaux d'apprentissage est généralement aléatoire pour ne pas influencer le chemin d'optimisation de façon déterministe. Avec un formalisme d'optimisation, ces méthodes effectuent l'approximation parcimonieuse de chaque signal y_p :

$$\min_{x_p} \| y_p - \Phi x_p \|_2^2 \quad \text{t.q.} \quad \|x_p\|_0 \leq K \,, \tag{1.21}$$

suivie immédiatement de la mise à jour du dictionnaire. La différence entre les deux types de mise à jour du dictionnaire est illustrée sur les Algorithmes 4 et 5. En outre, la mise à jour *online* peut être réalisée de deux manières, soit en utilisant uniquement le signal courant y_p [AE08] :

$$\min_{\Phi} \| y_p - \Phi x_p \|_2^2 \quad \text{t.q.} \quad \forall m \in \mathbb{N}_M, \|\phi_m\|_2 = 1 \,, \tag{1.22}$$

soit en utilisant les p signaux $\{y_\pi\}_{\pi=1}^p$ déjà décomposés [MBPS10, SE10] :

$$\min_{\Phi} \sum_{\pi=1}^{p} f \cdot \| y_\pi - \Phi x_\pi \|_2^2 \quad \text{t.q.} \quad \forall m \in \mathbb{N}_M, \|\phi_m\|_2 = 1 \,, \tag{1.23}$$

avec f un facteur pouvant dépendre de π ou de p.

La descente de gradient stochastique du 1er ordre utilisé dans [AE08] fournit un algorithme d'apprentissage réclamant peu de mémoire et de calculs par rapport aux algorithmes de type *batch*. Dans [BB08], Bottou et Bousquet expliquent que dans un processus itératif, chaque étape n'a pas besoin d'être minimisée parfaitement pour atteindre la solution espérée. Ils proposent ainsi l'utilisation de méthodes stochastiques. Fort de cela, Mairal *et al.* introduisent un apprentissage *online* utilisant un gradient stochastique et montrent des gains de rapidité pour de grandes bases de données comme pour des petites [MBPS10]. Notons aussi qu'ils utilisent une relaxation ℓ_1 pour la contrainte de parcimonie en employant le LARS. Une version *online* du ILS-DLA, appelée *Recursive Least-Squares* DLA (RLS-DLA), est présentée dans [SE10] avec également de meilleures performances que le ILS-DLA.

Si les méthodes *online* ne peuvent pas garantir la stricte décroissance de l'erreur quadratique globale, la nature stochastique de l'optimisation leur évite en général de tomber dans des minima locaux comme parfois constatés dans les méthodes *batch*.

TABLE 1.1 – Tableau résumé des principaux DLAs.

Nom	Approximation parcimonieuse	Mise à jour	Type
Approches probabilistes	Descentes de gradient puis approximations parcimonieuses	Descente de gradient	*batch*
MOD [EAH00] puis ILS-DLA [ESH07]	OMP, ORMP, FOCUSS	Moindres carrés	*batch*
K-SVD [AEB06a]	OMP	SVD	*batch*
online DLA [MBPS10]	LARS	Descente de gradient	*online*
RLS-DLA [SE10]	OMP	Moindres carrés	*online*

La Table 1.1 résume les principaux DLAs et leurs éléments constitutifs. Notons que les approximations parcimonieuses utilisées dans les DLAs sont facilement changeables et, dans les faits, facilement changées d'une application à l'autre.

Autres méthodes

Comme nous ne pouvons pas citer toutes les méthodes, nous évoquons seulement quelques méthodes ayant des formulations et des critères différents. Les méthodes les plus connues sont la méthode fondée sur la géométrie de la solution exacte du problème ℓ_1 [Plu07] ; la méthode dont la mise à jour du dictionnaire utilise une résolution duale [LBRN07] ; la méthode *Majorization-Minimization* [YBD09] qui remplace le critère d'optimisation par une fonction majorante plus facile à minimiser ; la méthode *Iterative Subspace Identification* (ISI) [GT10] qui enlève de l'ensemble d'apprentissage les signaux dont les atomes d'intérêt ont déjà été extraits.

Notons aussi que certaines méthodes d'apprentissage de dictionnaire sont regroupées sous le terme *dictionary design* à cause de la connaissance *a priori* qu'elles introduisent via des contraintes spécifiques. Nous citons le cas où le dictionnaire est vu comme l'union de bases [LGBB05], avec une contrainte d'orthogonalité entre les atomes de chaque base ; la contrainte de décorrélation des atomes [JVLG06, YDD09], la contrainte de non-négativité [DF12]. Le cas d'invariance par translation est parfois considéré comme appartenant à cette catégorie dû au fait que les atomes sont des noyaux paramétrés.

1.3.3 Remarques

Liens avec d'autres approches

Pour l'apprentissage d'une base, *i.e.* pour $\Phi \in \mathbb{R}^{N \times M}$ avec $M = N$, l'approche *Expectation-Maximization* (EM) [Moo96] alterne entre une étape *expectation* qui estime les coefficients de décomposition sur cette base, et une étape *maximization* qui optimise les vecteurs. L'apprentissage de dictionnaire peut être vu comme une extension de cette approche aux dictionnaires [TF11], où l'étape *expectation* est désormais une approximation parcimonieuse pour faire face à la redondance de la matrice Φ.

La K-SVD, et plus généralement l'apprentissage de dictionnaire, peut aussi être vue comme une généralisation de l'algorithme non-supervisé des K-*means* [AEB06a, SS09, TF11]. Dans notre formalisme, les K barycentres des classes correspondent aux M atomes du dictionnaire, et le problème des K-*means* s'écrit :

$$\min_{\Phi,X} \| Y - \Phi X \|_F^2 \quad \text{t.q.} \quad \forall p \in \mathbb{N}_P, x_p = \mathbb{1}_m \,, \tag{1.24}$$

avec $\mathbb{1}_m \in \mathbb{R}^M$ défini comme un vecteur de M zéro, exceptée sa $m^{\text{ième}}$ valeur égale à 1. Ainsi, au lieu d'attribuer chaque signal à un unique vecteur ϕ_m appris comme le barycentre de la classe, l'apprentissage de dictionnaire formalisé dans l'Eq. (1.16) attribue chaque signal à une combinaison linéaire parcimonieuse d'atomes, *i.e.* à un hyperplan de dimension K.

A propos de la mise à jour des atomes

Remarquons enfin que, indépendamment du type de mise à jour détaillé ci-dessus (*batch* / *online*), deux procédures de mise à jour sont possibles. A l'itération i, il existe :

- une mise à jour où tous les atomes du dictionnaire Φ^i sont mis à jour en utilisant le dictionnaire Φ^{i-1} de l'itération précédente et X^i, tels que ILS et RLS-DLA ;
- une mise à jour où les atomes de Φ^i déjà mis à jour sont utilisés pour mettre à jour ceux restants. Par exemple, dans la K-SVD où les atomes et les coefficients sont mis à jour simultanément, les éléments $\{\phi_\nu^i\}_{\nu=1}^m$ et $\{x_\nu^i\}_{\nu=1}^m$ déjà mis à jour sont utilisés pour mettre à jour ϕ_{m+1}^{i-1} et x_{m+1}^{i-1}.

1.3.4 DLA invariants par translation

Certains DLAs sont étendus au cas d'invariance par translation (cf. Section 1.1.3), réalisant l'apprentissage d'un dictionnaire de noyaux. A cause de la forte cohérence d'un tel dictionnaire, tous ces DLAs utilisent des poursuites ℓ_0 pour l'étape d'approximation parcimonieuse.

Les méthodes probabilistes sont étendues, avec :

1. une étape d'approximation parcimonieuse invariante par translation où les coefficients x sont calculés pour chaque signal, par descente de gradient [Ols00, Ols01] ou par approximation parcimonieuse [LS99, SL05b, SL06],
2. une mise à jour qui moyenne les translations actives d'un même noyau, par descente de gradient, puis normalisation des noyaux.

D'autres travaux ont proposé des méthodes invariantes par translation [WEK03, PABD06, BD06, GRKN07, MSH08], et notamment MoTIF (*Matching of Time Invariant Filters*) [JVLG06] qui intègre une contrainte de décorrélation des atomes lors de la mise à jour par SVD.

Les autres méthodes principales décrites en Section 1.3.2 sont étendues au cas d'invariance par translation en prenant en compte les cas de recouvrements d'atomes lors de la mise à jour des noyaux. Le MOD est étendu dans [AHSE01, SHA06] puis le ILS-DLA dans [ESH07] en prenant un facteur de translation égal à 1. La mise à jour est faite par moindres carrés incluant une matrice de recouvrements des atomes. La K-SVD est étendue dans [Aha06, Chap. 6] sous le nom *shift-invariant* K-SVD (SI-K-SVD) : sa mise à jour est aussi réalisée par moindres carrés avec une matrice de décalage. Une variante de la K-SVD invariante par translation est aussi étudiée dans [MLG+08], avec une mise à jour par SVD comprenant un terme supplémentaire en cas de recouvrements qui divise le résidu en parts égales entre les noyaux.

Pour conclure, dans cette section, nous avons expliqué l'apprentissage de dictionnaire et nous avons fait la revue des différents algorithmes résolvant ce problème. Les méthodes d'approximation parcimonieuse et d'apprentissage de dictionnaire sont sensibles à la cohérence du dictionnaire. Dans la section suivante, nous nous intéresserons à ce point.

1.4 Considérations théoriques sur la cohérence

Dans cette section, nous nous intéressons à la cohérence du dictionnaire, et plus particulièrement dans le cas d'invariance par translation. La cohérence mutuelle μ_Φ du dictionnaire normé $\Phi \in \mathbb{R}^{N\times M}$ est définie comme le maximum des produits scalaires entre atomes [Tro04] :

$$\mu_\Phi = \max_{i\neq j,\ i\in\mathbb{N}_M,\ j\in\mathbb{N}_M} |\langle \phi_i, \phi_j \rangle| \ . \tag{1.25}$$

Pour une base orthonormale, la cohérence mutuelle est nulle, et pour tout dictionnaire redondant, $1/\sqrt{N} \leq \mu_\Phi \leq 1$. Un dictionnaire est dit incohérent si μ_Φ est suffisamment faible. Cette valeur permet de qualifier le dictionnaire et les performances de décompositions parcimonieuses effectuées sur ce dernier.

1.4.1 Cas classique

De nombreux articles ont étudié [BDE09] l'influence du dictionnaire sur le problème de décomposition parcimonieuse, et sur le fait que le support du signal soit identifiable. Plusieurs conditions suffisantes sur la cohérence Φ et la parcimonie K de la solution sont données [DH01, BE02, GN03] pour avoir l'équivalence entre les différentes formulations énoncées en Section 1.2.1, ainsi que la stabilité à un bruit borné [DET06, Tro06] et la robustesse à la compressibilité. D'autres conditions permettent d'assurer de telles propriétés, utilisant l'isométrie restreinte [CT05], le support via l'*Exact Recovery Coefficient* [Fuc04, Tro06] ou encore le *spark* [DET06].

L'inconvénient de l'ensemble de ces critères est qu'ils sont très sévères afin de borner le pire des cas : une minimisation ℓ_1 est assurée d'identifier un signal s'il est suffisamment parcimonieux sur un dictionnaire incohérent. En particulier, l'équivalence entre les minimisations ℓ_0 (1.5) et ℓ_1 (1.7) est vérifiée si [DET06] :

$$\|x\|_0 \leq \frac{1}{4}\cdot\left(1+\frac{1}{\mu_\Phi}\right) \ . \tag{1.26}$$

Ces fortes hypothèses sur la cohérence du dictionnaire sont cependant difficilement vérifiables pour des applications réelles utilisant des dictionnaires appris. C'est encore plus difficile pour les dictionnaires

invariants par translation, très cohérents par nature. La cohérence de tels dictionnaires n'ayant pas fait l'objet d'études, nous allons chercher à la caractériser.

Remarquons aussi que ces considérations ont été récemment étendues au contexte de l'apprentissage de dictionnaire [AEB06b, GS10, GWW11, JGB12]. De même, les hypothèses pour identifier un dictionnaire sont très sévères.

1.4.2 Cas invariant par translation

Dans le cas de l'invariance par translation, le dictionnaire Φ s'avère être la concaténation de matrices de Toeplitz (cf. Section 1.1.3). La cohérence mutuelle d'un tel dictionnaire est très forte, souvent supérieure à 0.99 dans les applications réelles. Les différentes conditions évoquées précédemment ne sont donc pas vérifiables. Notons toutefois que des travaux récents ont étudié ces conditions [HBRN10, PRT12], mais pour des noyaux aléatoires, fournissant ainsi des estimations probabilistes des bornes d'isométrie restreinte.

Nous cherchons donc à caractériser plus finement la cohérence d'un dictionnaire dans le cas d'invariance par translation. Le produit scalaire entre un noyau $\psi(t)$ et sa translatée $\psi(t+1)$ dépend du contenu spectral du noyau : si ψ est un noyau de bruit blanc, le produit scalaire sera faible, alors que s'il est quasi-constant, il sera élevé. La proposition suivante formalise cette considération.

Proposition

Nous considérons un noyau réel ψ :

- *invariant par translation,*
- *d'énergie finie et normé, i.e. $\|\psi\|_2 = 1$,*
- *de support fini,*
- *de bande spectrale limitée*[3] *: sa fréquence réduite maximale est notée $\lambda_{max} \leq 1/2$ (cf. Fig. 1.8),*

alors nous avons :

$$1 - 2\pi^2 \cdot {\lambda_{max}}^2 \;\leq\; |\, \langle\, \psi(t), \psi(t+1)\,\rangle\, | \;\leq\; 1\,. \tag{1.27}$$

Dans le cas d'un dictionnaire Φ invariant par translation généré à partir de L noyaux $\{\psi_l\}_{l=1}^{L}$ (cf. Section 1.1.3), nous avons :

$$\max_{l \in \mathbb{N}_L} |\, \langle\, \psi_l(t), \psi_l(t+1)\,\rangle\, | \;\leq\; \mu_\Phi\,. \tag{1.28}$$

Le résultat de l'Eq. (1.27) permet donc de minorer la cohérence μ_Φ d'un dictionnaire invariant par translation.

Preuve

Nous considérons un signal réel discret $\psi(n)$, composé de N échantillons. Nous considérons la transformée de Fourier de signaux discrets (TFd) qui fournit en fréquence un spectre continu et périodique

3. La transformée de Fourier d'un signal de support fini est infinie. Cependant, ici, nous considérerons comme nulles les valeurs du spectre en dessous d'un certain seuil.

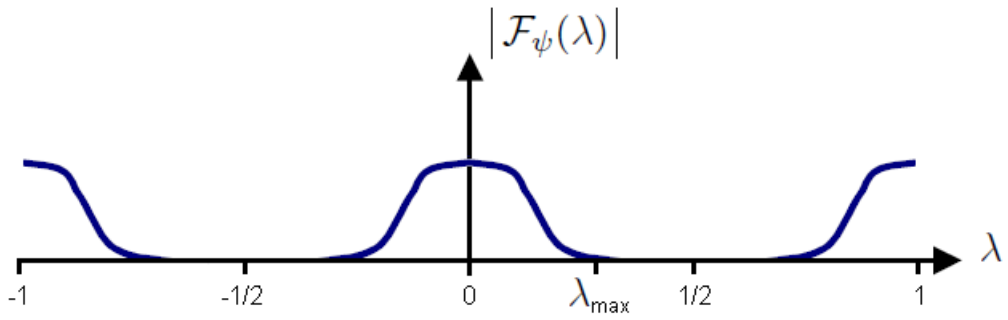


Fig. 1.8 – Illustration du spectre $\mathcal{F}_\psi(\lambda)$ du noyau ψ, considéré limité par la fréquence λ_{max}.

[Mal09]. Comme nous étudions les échantillons de ψ, nous travaillerons avec des fréquences réduites notées λ. La transformée de Fourier du signal ψ est :

$$\mathcal{F}_\psi(\lambda) = \sum_{n=1}^{N} \psi(n) \cdot e^{-2\pi \mathrm{i} \lambda n} \,, \tag{1.29}$$

avec le produit scalaire associé défini par :

$$\langle\, \mathcal{F}(\lambda), \mathcal{G}(\lambda) \,\rangle = \int_{-1/2}^{+1/2} \mathcal{F}^*(\lambda)\, \mathcal{G}(\lambda) d\lambda \,. \tag{1.30}$$

La transformée inverse est :

$$\psi(n) = \int_{-1/2}^{+1/2} \mathcal{F}_\psi(\lambda) \cdot e^{2\pi \mathrm{i} \lambda n} d\lambda \,. \tag{1.31}$$

Grâce à la conservation du produit scalaire, nous avons :

$$\begin{aligned}
\langle\, \psi(n), \psi(n+1) \,\rangle &= \left\langle\, \mathcal{F}_\psi(\lambda), \mathcal{F}_\psi(\lambda) \cdot e^{2\pi \mathrm{i} \lambda} \,\right\rangle && (1.32)\\
&= \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot e^{2\pi \mathrm{i} \lambda} d\lambda && (1.33)\\
&= \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \, d\lambda + \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot \left(e^{2\pi \mathrm{i} \lambda} - 1 \right) d\lambda && (1.34)\\
&= 1 + \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot \left(e^{2\pi \mathrm{i} \lambda} - 1 \right) d\lambda \,, \quad \text{grâce au théorème de Parseval.} && (1.35)
\end{aligned}$$

En réarrangeant les membres (1.33) et (1.35), nous avons :

$$\begin{aligned}
1 - \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot e^{2\pi \mathrm{i} \lambda} d\lambda &= \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot \left(1 - e^{2\pi \mathrm{i} \lambda} \right) d\lambda && (1.36)\\
&= \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot (1 - \cos(2\pi\lambda))\, d\lambda \quad \text{car } |\mathcal{F}_\psi(\lambda)| \text{ est paire,} && (1.37)\\
&\leq \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot \frac{(2\pi\lambda)^2}{2} d\lambda \quad \text{car } 1 - \cos(\theta) \leq \theta^2/2 \text{ (cf. Annexe 7.1),} && (1.38)\\
&\leq \frac{(2\pi\lambda_{max})^2}{2} \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \, d\lambda = \frac{(2\pi\lambda_{max})^2}{2} \,. && (1.39)
\end{aligned}$$

En considérant la valeur absolue de l'Eq. (1.36), nous avons :

$$\left| 1 - \left| \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot e^{2\pi \mathrm{i} \lambda} d\lambda \right| \right| \leq \left| 1 - \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot e^{2\pi \mathrm{i} \lambda} d\lambda \right| \leq 2\pi^2 {\lambda_{max}}^2 . \tag{1.40}$$

Au final, nous avons :

$$1 - 2\pi^2 {\lambda_{max}}^2 \leq \left| \int_{-1/2}^{+1/2} |\mathcal{F}_\psi(\lambda)|^2 \cdot e^{2\pi \mathrm{i} \lambda} d\lambda \right| = | \langle \psi(n), \psi(n+1) \rangle | . \tag{1.41}$$

Et, par l'inégalité de Cauchy-Schwarz, nous obtenons l'inégalité de droite :

$$| \langle \psi(n), \psi(n+1) \rangle | \leq \|\psi(n)\|_2 \cdot \|\psi(n+1)\|_2 = 1 . \tag{1.42}$$

□

Interprétation

D'une part, la minoration de l'Eq. (1.27) prouve que la cohérence d'un dictionnaire invariant par translation est fonction du contenu spectral de ses noyaux. Quand le noyau est basse fréquence, *i.e.* $\lambda_{max} \to 0$, le produit scalaire entre ce noyau $\psi(t)$ et sa translatée $\psi(t+1)$ tend vers 1. Nous remarquons que quand λ_{max} devient grand, la borne reste vraie mais perd de son intérêt, et *a fortiori* au-delà de $\lambda = 1/\sqrt{2}\pi \approx 0.23$ où la borne devient négative.

D'autre part, nous constatons que le produit scalaire entre un noyau de bruit blanc et sa translatée est très faible (théoriquement nul), alors que ce noyau ne contient aucune information. C'est pour cela qu'il est utilisé comme initialisation de l'apprentissage de dictionnaire, car il n'apporte aucun *a priori*. A la fin de l'apprentissage, quand les noyaux ont convergé vers des motifs caractéristiques contenant de l'information, *i.e.* vers des noyaux basses fréquences, le produit scalaire en question est très élevé.

Ainsi, la cohérence d'un dictionnaire invariant par translation dont les noyaux sont informatifs est par nature très élevée. Les algorithmes très sensibles à la cohérence mutuelle du dictionnaire ne sont donc pas utilisables.

Application

Nous appliquons ce résultat à un dictionnaire de sinus cardinaux. Nous définissons le noyau théorique ψ comme :

$$\psi_{th}(t) = \mathrm{sinc}(Ct) = \frac{\sin(\pi\, C\, t)}{\pi\, C\, t} . \tag{1.43}$$

La transformée de Fourier de ψ_{th} est un créneau défini comme :

$$\mathcal{F}_{\psi_{th}}(\lambda) = \frac{1}{C} \mathrm{rect}(\lambda / C) \ = \begin{cases} 1/C & \text{pour } |\lambda| \leq C/2 \\ 0 & \text{sinon} \end{cases} . \tag{1.44}$$

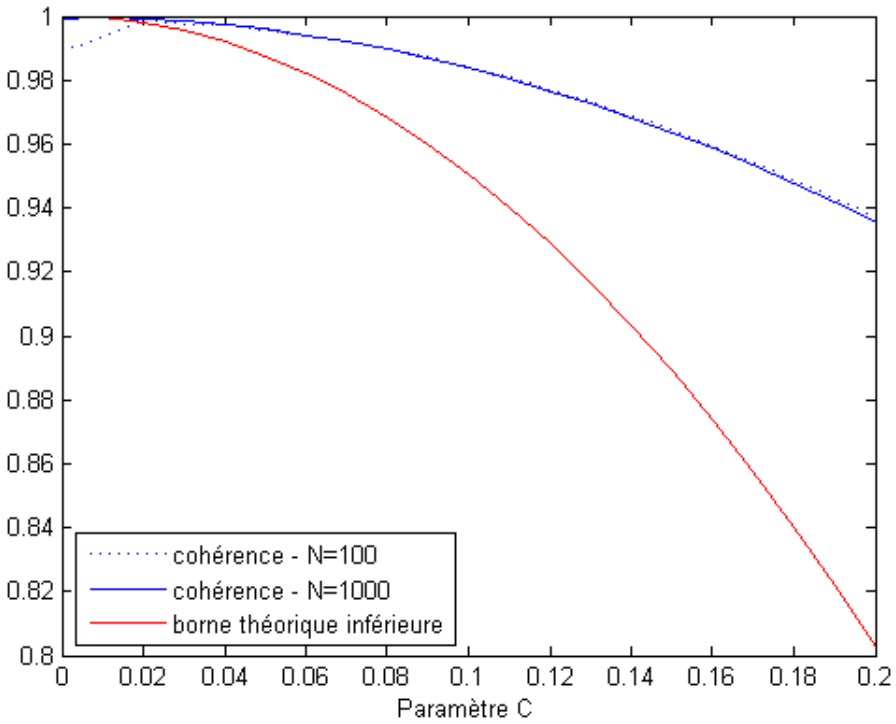


Fig. 1.9 – Valeurs des cohérences des dictionnaires et de la borne théorique en fonction du paramètre C.

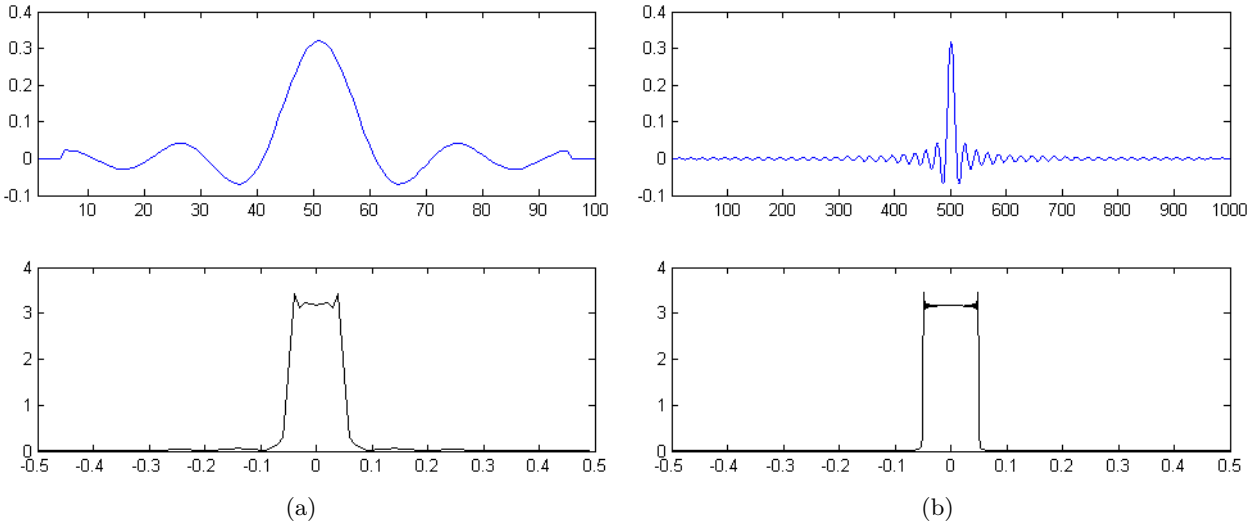
Théoriquement, la fréquence maximale λ_{max} de ψ_{th} est donc égale à $C/2$. D'autre part, le noyau ψ_{th} est normé. Grâce au théorème de Parseval, nous avons :

$$\|\psi_{th}\|_2 = \|\mathcal{F}_{\psi_{th}}\|_2 = \left(\frac{1}{C} \cdot \int_{-C/2}^{+C/2} 1 d\lambda\right)^{1/2} = \left(\frac{1}{C} \cdot C\right)^{1/2} = 1\,. \tag{1.45}$$

Ce noyau ψ_{th} est théorique, car pour une implémentation numérique, le noyau doit être à support fini. Nous considérerons aussi que dans cet exemple mono-noyau ($L = 1$), l'Eq. (1.28) est une égalité, *i.e.* le produit scalaire $|\langle \psi(t), \psi(t+1) \rangle|$ est égal à la cohérence de ce dictionnaire.

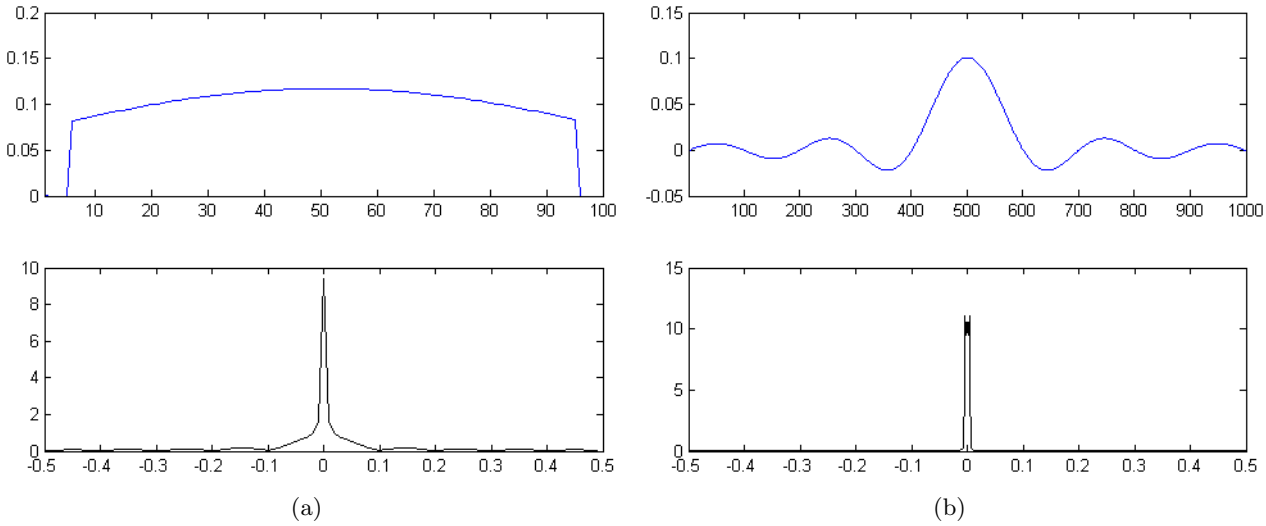
Nous appelons ψ le noyau à support fini. L'une des conséquences est que sa transformée de Fourier est infinie. Cet effet de bord impactera les résultats pour des noyaux courts. Pour notre application, nous choisissons donc un noyau court de longueur $N = 100$ et un noyau long de longueur $N = 1000$. L'autre conséquence est que ce noyau ψ n'est plus normé. Une étape de normalisation est donc nécessaire pour générer ensuite le dictionnaire normé invariant par translation. Nous faisons varier le paramètre C qui dilate temporellement le noyau étudié. Nous traçons en Fig. 1.9 la valeur de la cohérence du dictionnaire pour chacun des noyaux, *i.e.* le produit scalaire $\langle \psi(t), \psi(t+1) \rangle$, ainsi que la valeur inférieure de la borne théorique de l'Eq. (1.27) en fonction du paramètre C. Nous observons que les résultats sont conformes à la borne annoncée, sauf quand les valeurs N et C sont faibles. Il s'agit en réalité de l'effet de bord dû à l'implémentation numérique des noyaux à support fini.

Pour mieux visualiser ce phénomène, nous traçons les deux noyaux ψ pour $C = 0.1$ et $C = 0.01$, ainsi que leurs spectres (en valeurs absolues). Pour $C = 0.1$, où l'effet de bord n'est pas visible, nous traçons le noyau avec $N = 100$ en Fig. 1.10(a) et celui avec avec $N = 1000$ en Fig. 1.10(b). Nous observons que le noyau court avec $N = 100$ possède un spectre dont la fréquence réduite maximale λ_{max} est moins nette



(a) (b)

Fig. 1.10 – Noyaux de sinus cardinal (haut) et leurs spectres (bas) pour $C = 0.1$, et de longueur $N = 100$ (a) et $N = 1000$ (b).



(a) (b)

Fig. 1.11 – Noyaux de sinus cardinal (haut) et leurs spectres (bas) pour $C = 0.01$, et de longueur $N = 100$ (a) et $N = 1000$ (b).

que celle du noyau long. Cependant, les deux noyaux vérifient à peu près les conditions d'application de l'Eq. (1.27).

Pour $C = 0.01$, où l'effet de bord est très visible, nous traçons le noyau avec $N = 100$ en Fig. 1.11(a) et celui avec avec $N = 1000$ en Fig. 1.11(b). Alors que le noyau long avec $N = 1000$ possède une fréquence maximale à peu près nette, ce n'est plus le cas du noyau court avec $N = 100$. Ces deux noyaux sont plus dilatés que les deux précédents à cause du paramètre C. Ainsi, quand le noyau ψ_{th} est écourté sur $N = 100$ échantillons, il devient très déformé (cf. Fig. 1.11(a) (haut)), et ne ressemble plus à un sinus cardinal. En conséquence, sa transformée de Fourier est déformée elle aussi (cf. Fig. 1.11(a) (haut)) et possède une queue lourde, ce qui l'empêche de vérifier les conditions d'application de l'Eq. (1.27). Cet

effet est logiquement atténué quand $N = 1000$.

Dans cette section, nous nous sommes intéressés à la cohérence dans le cas d'invariance par translation. Nous avons proposé une borne inférieure pour la cohérence d'un dictionnaire invariant par translation, démontrant qu'elle est élevée par nature. De plus, nous avons appliqué numériquement ces considérations théoriques sur un dictionnaire composé de sinus cardinaux. La borne proposée est effectivement respectée, aux effets de bord près.

Conclusion

Dans son ouvrage L'Art poétique [Boi74], Nicolas Boileau disait :

Ce que l'on conçoit bien s'énonce clairement,
Et les mots pour le dire arrivent aisément.

Pour résumer ce chapitre d'introduction aux représentations parcimonieuses, nous pourrions transposer cette citation en :

Ce que l'on conçoit bien se représente parcimonieusement,
Car les atomes pour le dire s'apprennent aisément.

L'apprentissage de dictionnaire, alternant entre une approximation et une mise à jour, est une méthode efficace pour apprendre un dictionnaire adapté à une base de données. Ce dictionnaire a la particularité de fournir des représentations parcimonieuses aux signaux de cette base. Il peut ensuite être utilisé pour des traitements tels que le débruitage, la super-résolution, l'*inpainting*, etc. Ce dictionnaire est aussi la collection des motifs informatifs, répétitifs et énergétiques extraits des données. Il permet donc de comprendre les strutures élémentaires d'une base de données. Nous avons aussi considéré le cas particulier de l'invariance par translation utile pour étudier les signaux temporels.

Dans la suite de la thèse, cette approche va être étendue pour traiter des signaux multivariés composés de plusieurs canaux acquis simultanément. Nous étudierons aussi le cas où ces différents canaux auront des interactions entre eux, notamment dans le cadre de la rotation.

Bibliographie

[AE08] M. Aharon et M. Elad : Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM J. Imaging Sciences*, 1:228–247, 2008.

[AEB06a] M. Aharon, M. Elad et A.M. Bruckstein : K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54:4311–4322, 2006.

[AEB06b] M. Aharon, M. Elad et A.M. Bruckstein : On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Linear Algebra and its Applications*, 416:48–67, 2006.

[Aha06] M. Aharon : *Overcomplete Dictionaries for Sparse Representation of Signals.* Thèse de doctorat, Technion - Israel Institute of Technology, 2006.

[AHSE01] S.O. Aase, J.H. Husøy, K. Skretting et K. Engan : Optimized signal expansions for sparse representation. *IEEE Trans. on Signal Processing*, 49:1087–1096, 2001.

[BB08] L. Bottou et O. Bousquet : The tradeoffs of large scale learning. *In Advances in Neural Information Processing Systems NIPS*, volume 20, pages 161–168, 2008.

[BBC11] S. Becker, J. Bobin et E.J. Candès : NESTA : A fast and accurate first-order method for sparse recovery. *SIAM J. Imaging Sciences*, 4:1–39, 2011.

[BD06] T. Blumensath et M.E. Davies : Sparse and shift-invariant representations of music. *IEEE Trans. on Audio, Speech, and Language Processing*, 14:50–57, 2006.

[BD07] T. Blumensath et M.E. Davies : On the difference between orthogonal matching pursuit and orthogonal least squares. Rapport technique, Edinburgh University, 2007.

[BD08] T. Blumensath et M.E. Davies : Gradient pursuits. *IEEE Trans. on Signal Processing*, 56:2370–2382, 2008.

[BD09] T. Blumensath et M.E. Davies : Iterative hard thresholding for compressed sensing. *Appl. Comput. Harmon. Anal.*, 27:265–274, 2009.

[BD10] T. Blumensath et M.E. Davies : Normalized iterative hard thresholding : Guaranteed stability and performance. *IEEE Journal of Selected Topics in Signal Processing*, 4:298–309, 2010.

[BDE09] A.M. Bruckstein, D.L. Donoho et M. Elad : From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51:34–81, 2009.

[BDF07] J.M. Bioucas-Dias et M.A.T. Figueiredo : A new TwIST : Two-step iterative shrinkage/ thresholding algorithms for image restoration. *IEEE Trans. on Image Processing*, 16:2992–3004, 2007.

[BE02] A.M. Bruckstein et M. Elad : A generalized uncertainty principle and sparse representation in pairs of bases. *IEEE Trans. on Information Theory*, 48:2558–2567, 2002.

[Blu05] T. Blumensath : *Bayesian Modelling of Music : Algorithmic Advances and Experimental Studies of Shift-Invariant Sparse Coding.* Thèse de doctorat, University of London, 2005.

[Boi74] N. Boileau : *L'Art poétique.* 1674.

[BT09] A. Beck et M. Teboulle : A fast iterative shrinkage-threshold algorithm for linear inverse problems. *SIAM J. Imaging Sciences*, 2:183–202, 2009.

[CARKD99] S.F. Cotter, R. Adler, R.D. Rao et K. Kreutz-Delgado : Forward sequential algorithms for best basis selection. *IEE Proceedings - Vision, Image and Signal Processing*, 146:235–244, 1999.

[CBL89] S. Chen, S.A. Billings et W. Luo : Orthogonal least squares methods and their application to non-linear system identification. *Int. Journal of Control*, 50:1873–1896, 1989.

[CD99] E.J. Candès et D. Donoho : Ridgelets : The key to higher-dimensional intermittency ? *Phil. Trans. R. Soc. Lond. A.*, 357:2495–2509, 1999.

[CD00] E.J. Candès et D.L. Donoho : *Curvelets - A Surprisingly Effective Nonadaptive Representation For Objects with Edges*, pages 105 – 120. Curves and Surfaces, Vanderbilt University Press, 2000.

[CDS98] S. Chen, D.L. Donoho et M.A. Saunders : Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20:33–61, 1998.

[CJ10] P. Comon et C. Jutten : *Handbook of Blind Source Separation, Independent Component Analysis and Applications*. New York : Academic, 2010.

[Com94] P. Comon : Independent Component Analysis, a new concept ? *Signal Process.*, 36:287–314, 1994.

[CR05] E. Candès et J. Romberg : ℓ_1-MAGIC : Recovery of sparse signals via convex programming. Rapport technique, California Inst. Technol., 2005.

[CT05] E.J. Candès et T. Tao : Decoding by linear programming. *IEEE Trans. on Information Theory*, 51:4203–4215, 2005.

[CW95] S. Chen et J. Wigger : Fast orthogonal least squares algorithm for efficient subset model selection. *IEEE Trans. on Signal Processing*, 43:1713–1715, 1995.

[CW05] P.L. Combettes et V.R. Wajs : Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4:1168–1200, 2005.

[CWB08] E. Candès, M.B. Wakin et S.P. Boyd : Enhancing sparsity by reweighted ℓ_1 minimization. *J. Fourier Anal. Appl.*, 14:877–905, 2008.

[Dav94] G. Davis : *Adaptive Nonlinear Approximations*. Thèse de doctorat, New York University, 1994.

[DDDM04] I. Daubechies, M. Defrise et C. De Mol : An iterative algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.*, LVII:1413–1457, 2004.

[DET06] D.L. Donoho, M. Elad et V. Temlyakov : Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. on Information Theory*, 52:6–18, 2006.

[DF12] O. Dikmen et C. Févotte : Maximum marginal likelihood estimation for nonnegative dictionary learning in the Gamma-Poisson model. *IEEE Trans. on Signal Processing*, 60:5163–5175, 2012.

[DH01] D.L. Donoho et X. Huo : Uncertainty principle and ideal atomic decompositions. *IEEE Trans. on Information Theory*, 47:2845–2862, 2001.

[DM09] W. Dai et O. Milenkovic : Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. on Information Theory*, 55:2230–2249, 2009.

[DMZ94] G. Davis, S. Mallat et Z. Zhang : Adaptive time-frequency decompositions with matching pursuits. *Optical Engineering*, 33:2183–2191, 1994.

[DT08] D.L. Donoho et Y. Tsaig : Fast solution of ℓ_1-norm minimization problems when the solution may be sparse. *IEEE Trans. on Information Theory*, 54:4789–4812, 2008.

[DTDS06] D.L. Donoho, Y. Tsaig, I. Drori et J.L. Starck : Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. Rapport technique, Stanford University, 2006.

[DVT96] R.A. De Vore et V.N. Temlyakov : Some remarks on greedy algorithms. *Advances in Computational Mathematics*, 5:173–187, 1996.

[EA06] M. Elad et M. Aharon : Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. on Image Processing*, 15:3736–3745, 2006.

[EAH00] K. Engan, S.O. Aase et J.H. Husøy : Multi-frame compression : theory and design. *Signal Process.*, 80:2121–2140, 2000.

[EHJT04] B. Efron, T. Hastie, I. Johnstone et R. Tibshirani : Least Angle Regression. *Ann. Stat.*, 32:407–499, 2004.

[Ela06] M. Elad : Why simple shrinkage is still relevant for redundant representations. *IEEE Trans. on Information Theory*, 52:5559–5569, 2006.

[ESH07] K. Engan, K. Skretting et J.H. Husøy : Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation. *Digit. Signal Process.*, 17:32–49, 2007.

[FNW07] M.A.T. Figueiredo, R.D. Nowak et S.J. Wright : Gradient projection for sparse reconstruction : Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics in Signal Processing*, 1:586–597, 2007.

[FSBM10] M.J. Fadili, J.-L. Starck, J. Bobin et Y. Moudden : Image decomposition and separation using sparse representations : An overview. *Proceedings of the IEEE*, 98:983–994, 2010.

[Fuc04] J.-J. Fuchs : On sparse representations in arbitrary redundant bases. *IEEE Trans. on Information Theory*, 50:1341–1344, 2004.

[GN03] R. Gribonval et M. Nielsen : Sparse representations in unions of bases. *IEEE Trans. on Information Theory*, 49:3320–3325, 2003.

[GR97] I.F. Gorodnitsky et B. Rao : Sparse signal reconstruction from limited data using FOCUSS : A re-weighted minimum norm algorithm. *IEEE Trans. on Signal Processing*, 45:600–616, 1997.

[GRKN07] H. Grosse, R. Raina, R. Kwong et A.Y. Ng : Shift-invariant sparse coding for audio classification. *In Proc. Conf. Uncertainty in Artificial Intelligence UAI*, pages 149–158, 2007.

[GS10] R. Gribonval et K. Schnass : Dictionary identification — sparse matrix-factorization via ℓ_1-minimization. *IEEE Trans. on Information Theory*, 56:3523–3539, 2010.

[GT10] B.V. Gowreesunker et A.H. Tewfik : Learning sparse representation using iterative subspace identification. *IEEE Trans. on Signal Processing*, 58:3055–3065, 2010.

[GWW11] Q. Geng, H. Wang et J. Wright : On the local correctness of ℓ_1 minimization for dictionary learning. *Preprint arXiv :1101.5672*, 2011.

[HBRN10] J. Haupt, W.U. Bajwa, G. Raz et R. Nowak : Toeplitz compressed sensing matrices with applications to sparse channel estimation. *IEEE Trans. on Information Theory*, 56:5862–5875, 2010.

[HO00] A. Hyvärinen et E. Oja : Independent component analysis : algorithms and applications. *Neural Networks*, 13:411–430, 2000.

[HYZ08] E.T. Hale, W. Yin et Y. Zhang : A fixed-point continuation method for ℓ_1-minimization : Methodology and convergence. *SIAM J. Optim.*, 19:1107–1130, 2008.

[JDCP11] L. Jacques, L. Duval, C. Chaux et G. Peyré : A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity. *Signal Process.*, 91:2699–2730, 2011.

[JGB12] R. Jenatton, R. Gribonval et F. Bach : Local stability and robustness of sparse dictionary learning in the presence of noise. *Preprint arXiv :1210.0685*, 2012.

[JMH+00] T.-P. Jung, S. Makeig, C. Humphries, T.-W. Lee, M.J. McKeown, V. Iragui et T.J. Sejnowski : Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, 37:163–178, 2000.

[JVLG05] P. Jost, P. Vandergheynst, S. Lesage et R. Gribonval : Learning redundant dictionaries with translation invariance property : the MoTIF algorithm. *In Workshop Structure et parcimonie pour la représentation adaptative de signaux SPARS '05*, 2005.

[JVLG06] P. Jost, P. Vandergheynst, S. Lesage et R. Gribonval : MoTIF : An efficient algorithm for learning translation invariant dictionaries. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '06*, pages 857–860, 2006.

[KDMR+03] K. Kreutz-Delgado, J.F. Murray, B.D. Rao, K. Engan, T. Lee et T.J. Sejnowski : Dictionary learning algorithms for sparse representation. *Neural Comput.*, 15:349–396, 2003.

[KKL+07] S.J. Kim, K. Koh, M. Lustig, S. Boyd et D. Gorinevsky : An interior-point method for large-scale ℓ_1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1:606–617, 2007.

[LBRN07] H. Lee, A. Battle, R. Raina et A.Y. Ng : Efficient sparse coding algorithms. *In Advances in Neural Information Processing Systems NIPS*, pages 801–808, 2007.

[LGBB05] S. Lesage, R. Gribonval, F. Bimbot et L. Benaroya : Learning unions of orthonormal bases with thresholded singular value decomposition. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '05*, pages 293–296, 2005.

[LO99] M.S. Lewicki et B.A. Olshausen : A probabilistic framework for the adaptation and comparison of image codes. *Journal of the Optical Society of America A*, 16:1587–1601, 1999.

[LS98] M.S. Lewicki et T.J. Sejnowski : Learning overcomplete representations. *Neural Comput.*, 12:337–365, 1998.

[LS99] M.S. Lewicki et T.J. Sejnowski : Coding time-varying signals using sparse, shift-invariant representations. *In Advances in Neural Information Processing Systems II*, pages 730–736, 1999.

[Mal09] S. Mallat : *A Wavelet Tour of signal processing.* 3rd edition, New-York : Academic, 2009.

[MBPS10] J. Mairal, F. Bach, J. Ponce et G. Sapiro : Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010.

[MBZJ09] H. Mohimani, M. Babaie-Zadeh et C. Jutten : A fast approach for overcomplete sparse decomposition based on smoothed ℓ_0 norm. *IEEE Trans. on Signal Processing*, 57:289–301, 2009.

[MCW05] D.M. Malioutov, M. Cetin et A.S. Willsky : Homotopy continuation for sparse signal representation. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '05*, pages 733–736, 2005.

[MD10] A. Maleki et D.L. Donoho : Optimally tuned iterative reconstruction algorithms for compressed sensing. *IEEE Journal of Selected Topics in Signal Processing*, 4:330–341, 2010.

[MES08] J. Mairal, M. Elad et G. Sapiro : Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17:53–69, 2008.

[MGB+09] B. Mailhé, R. Gribonval, F. Bimbot, M. Lemay, P. Vandergheynst et J.M. Vesin : Dictionary learning for the sparse modelling of atrial fibrillation in ECG signals. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '09*, pages 465–468, 2009.

[MJV+07] G. Monaci, P. Jost, P. Vandergheynst, B. Mailhé, S. Lesage et R. Gribonval : Learning multimodal dictionaries. *IEEE Trans. on Image Processing*, 16:2272–2283, 2007.

[MLG+08] B. Mailhé, S. Lesage, R. Gribonval, F. Bimbot et P. Vandergheynst : Shift-invariant dictionary learning for sparse representations : Extending K-SVD. *In Proc. Eur. Signal Process. Conf. EUSIPCO '08*, 2008.

[Moo96] T.K. Moon : The Expectation-Maximization algorithm. *IEEE Signal Processing Magazine*, 13:47–60, 1996.

[MSH08] M. Mørup, M.N. Schmidt et L.K. Hansen : Shift invariant sparse coding of image and music data. Rapport technique, Technical University of Denmark, 2008.

[MVS09] G. Monaci, P. Vandergheynst et F.T. Sommer : Learning bimodal structure in audio-visual data. *IEEE Trans. on Neural Networks*, 20:1898–1910, 2009.

[MZ93] S.G. Mallat et Z. Zhang : Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, 41:3397–3415, 1993.

[NT09] D. Needell et J.A. Tropp : CoSaMP : Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.*, 26:301–321, 2009.

[NV09] D. Needell et R. Vershynin : Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.*, 9:317–334, 2009.

[NWY11] M.K. Ng, F. Wang et X. Yuan : Inexact alternating direction methods for image recovery. *SIAM J. Sci. Comput.*, 33:1643–1668, 2011.

[OF96] B.A. Olshausen et D.J. Field : Natural image statistics and efficient coding. *Network Comp. Neural Syst.*, 7:333–339, 1996.

[OF97] B.A. Olshausen et D.J. Field : Sparse coding with an overcomplete basis set : a strategy employed by V1 ? *Vision Research*, 37:3311–3325, 1997.

[Ols00] B.A. Olshausen : Sparse coding of time-varying natural images. *In Proc. Int. Conf. Independent Component Analysis and Blind Source Separation*, pages 603–608, 2000.

[Ols01] B.A. Olshausen : Sparse codes and spikes. *In Probabilistic models of the brain : Perception and Neural Function*, pages 257–272. MIT Press, 2001.

[OPT00] M.R. Osborne, B. Presnell et B.A. Turlach : A new approach to variable selection in least squares problems. *IMA Journal of Numerical Analysis*, 20:389–404, 2000.

[PABD06] M.D. Plumbley, S.A. Abdallah, T. Blumensath et M.E. Davies : Sparse representations of polyphonic music. *Signal Process.*, 86:417–431, 2006.

[Plu07] M.D. Plumbley : Dictionary learning for l1-exact sparse coding. *In ICA 2007*, volume 4666 de *Lecture Notes in Computer Science*, pages 406–413, 2007.

[PRK93] Y.C. Pati, R. Rezaiifar et P.S. Krishnaprasad : Orthogonal Matching Pursuit : recursive function approximation with applications to wavelet decomposition. *In Conf. Record of the Asilomar Conf. on Signals, Systems and Comput.*, pages 40–44, 1993.

[PRT12] G.E. Pfander, H. Rauhut et J.A. Tropp : The restricted isometry property for time-frequency structured random matrices. *Probab. Theory Relat. Fields*, 2012.

[RBE10] R. Rubinstein, A.M. Bruckstein et M. Elad : Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98:1045–1057, 2010.

[RKD99] B. Rao et K. Kreutz-Delgado : An affine scaling methodology for best basis selection. *IEEE Trans. on Signal Processing*, 47:187–200, 1999.

[RNL02] L. Rebollo-Neira et D. Lowe : Optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9:137–140, 2002.

[ROF92] L.I. Rudin, S. Osher et E. Fatemi : Nonlinear total variation based noise removal algorithms. *Physica D : Nonlinear Phenomena*, 60:259–268, 1992.

[Sau02] M.A. Saunders : PDCO : Primal-dual interior-point method for convex objectives. Rapport technique, Stanford University, 2002.

[SBT00] S. Sardy, A.G. Bruce et P. Tseng : Block coordinate relaxation methods for nonparametric wavelet denoising. *Journal of Computational and Graphical Statistics*, 9:361–379, 2000.

[SE10] K. Skretting et K. Engan : Recursive least squares dictionary learning algorithm. *IEEE Trans. on Signal Processing*, 58:2121–2130, 2010.

[SED04] J.-L. Starck, M. Elad et D.L. Donoho : Redundant multiscale transforms and their application for morphological component analysis. *Advances in Imaging and Electron Physics*, 132:287–348, 2004.

[SHA06] K. Skretting, J.H. Husøy et S.O. Aase : General design algorithm for sparse frame expansions. *Signal Process.*, 86:117–126, 2006.

[SIBD11] C. Soussen, J. Idier, D. Brie et J. Duan : From Bernoulli-Gaussian deconvolution to sparse signal restoration. *IEEE Trans. on Signal Processing*, 59:4572–4584, 2011.

[SJS08] R. Sameni, C. Jutten et M.B. Shamsollahi : Multichannel electrocardiogram decomposition using periodic component analysis. *IEEE Trans. on Biomedical Engineering*, 55:1935–1940, 2008.

[SL05a] E. Smith et M.S. Lewicki : Efficient coding of time-relative structure using spikes. *Neural Comput.*, 17:19–45, 2005.

[SL05b] E. Smith et M.S. Lewicki : Learning efficient auditory codes using spikes predicts cochlear filters. *In Advances in Neural Information Processing Systems NIPS*, pages 1289–1296, 2005.

[SL06] E. Smith et M.S. Lewicki : Efficient auditory coding. *Nature*, 439:978–982, 2006.

[SS09] A. Szlam et G. Sapiro : Discriminative k-metrics. *In Proc. Int. Conf. Machine Learning ICML '09*, pages 1009–1016, 2009.

[TF11] I. Tošić et P. Frossard : Dictionary learning. *IEEE Signal Processing Magazine*, 28:27–38, 2011.

[Tib96] R. Tibshirani : Regression shrinkage and selection via the LASSO. *J. R. Statist. Soc. B*, 58:267–288, 1996.

[Tro04] J.A. Tropp : Greed is good : algorithmic results for sparse approximation. *IEEE Trans. on Information Theory*, 50:2231–2242, 2004.

[Tro06] J.A. Tropp : Just relax : Convex programming methods for subset selection and sparse approximation. *IEEE Trans. on Information Theory*, 52:1030–1051, 2006.

[TW10] J.A. Tropp et S.J. Wright : Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98:948–958, 2010.

[VB02] P. Vincent et Y. Bengio : Kernel matching pursuit. *Machine Learning*, 48:165–187, 2002.

[VML04] V.D. Vrabie, J.I. Mars et J.-L. Lacoume : Modified singular value decomposition by means of independent component analysis. *Signal Process.*, 84:645–652, 2004.

[WEK03] H. Wersing, J. Eggert et E. Körner : Sparse coding with invariance constraints. *In Proc. Int. Conf. Artificial Neural Networks ICANN*, pages 385–392, 2003.

[WNF09] S.J. Wright, R.D. Nowak et M.A.T. Figueiredo : Sparse reconstruction by separable approximation. *IEEE Trans. on Signal Processing*, 57:2479–2493, 2009.

[YBD09] M. Yaghoobi, T. Blumensath et M.E. Davies : Dictionary learning for sparse approximations with the majorization method. *IEEE Trans. on Signal Processing*, 57:2178–2191, 2009.

[YBW85] R. Yarlagadda, J.B. Bednar et T.L. Watt : Fast algorithms for ℓ_p deconvolution. *IEEE Trans. on Accoust., Speech, and Signal Processing*, ASSP-33:174–182, 1985.

[YDD09] M. Yaghoobi, L. Daudet et M.E. Davies : Parametric dictionary design for sparse coding. *IEEE Trans. on Signal Processing*, 57:4800–4810, 2009.

[YWHM10] J. Yang, J. Wright, T.S. Huang et Y. Ma : Image super-resolution via sparse representation. *IEEE Trans. on Image Processing*, 19:2861–2873, 2010.

[YZ11] J. Yang et Y. Zhang : Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM J. Sci. Comput.*, 33:250–278, 2011.

[ZEP12] R. Zeyde, M. Elad et M. Protter : On single image scale-up using sparse-representations. *In Curves and Surfaces*, volume 6920 de *Lecture Notes in Computer Science*, pages 711–730, 2012.

[Zou05] T. Zou, H. Hastie : Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, 67:301–320, 2005.

Chapitre 2

Représentations parcimonieuses multivariées

Introduction

Avec l'amélioration technologique des capteurs, les signaux temps-séries possèdent souvent plusieurs composantes acquises simultanément. Par exemple, les signaux électroencéphalographiques sont le fruit de l'acquisition d'un signal temporel pour chaque électrode, les signaux de mouvements 3D possèdent une trame temporelle de coordonnées pour chaque dimension, etc. Dans ce chapitre, nous allons voir comment les méthodes de représentations parcimonieuses étudiées dans le chapitre précédent sont étendues à ces signaux multicomposantes.

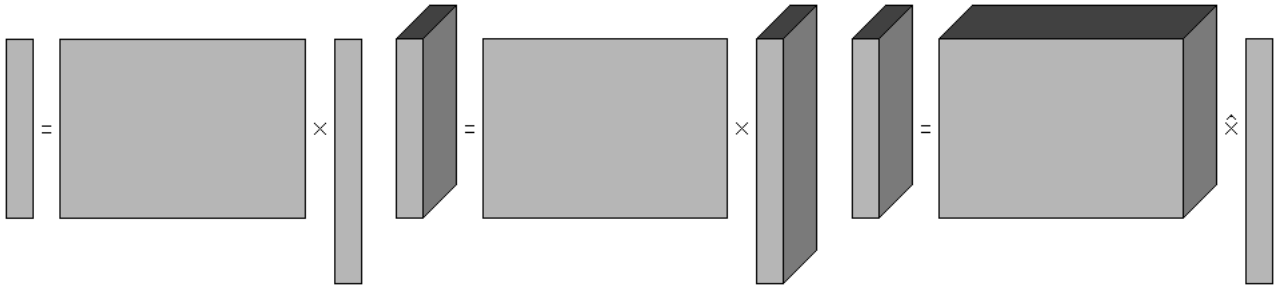
D'abord, nous allons introduire le modèle multivarié et le comparer au modèle multicanal très connu. Ensuite, nous détaillerons les extensions multivariées de l'OMP et du DLA. Nous comparerons notre algorithme d'apprentissage de dictionnaire aux autres algorithmes invariants par translation dans un cas univarié, et enfin, nous appliquerons nos méthodes sur des données réelles multicomposantes électroencéphalographiques (EEG).

2.1 Modèle multicanal versus multivarié

Dans le chapitre 1, nous avons travaillé avec un signal univarié $y \in \mathbb{R}^N$, et l'équation de cette approche univariée/monocanale est illustrée en Fig. 2.1(a). Dans le cas multicomposante, le signal étudié devient $\boldsymbol{y} \in \mathbb{R}^{N \times V}$, avec V le nombre de composantes [1] (aussi appelées canaux, variables ou sources). De plus, nous notons $\boldsymbol{y}[v] \in \mathbb{R}^N$ la $v^{\text{ième}}$ composante du signal $\boldsymbol{y}$. Dans la suite de cette thèse, nous noterons en gras toutes les variables de nature multicomposante. Pour ces données multicomposantes, deux modèles peuvent être envisagés selon la nature du dictionnaire utilisé :

- soit le dictionnaire $\Phi \in \mathbb{R}^{N \times M}$ est unicomposante et les coefficients $\boldsymbol{x} \in \mathbb{R}^{M \times V}$ sont multicomposantes, ce qui donne le modèle *multicanal* illustré en Fig. 2.1(b),
- soit le dictionnaire $\boldsymbol{\Phi} \in \mathbb{R}^{N \times M \times V}$ est multicomposante et les coefficients $x \in \mathbb{R}^M$ sont unicomposantes, ce qui donne le modèle *multivarié* illustré en Fig. 2.1(c).

1. Le terme *composante* utilisé dans la suite de la thèse aura désormais ce sens.



(a) $\hat{y} = \Phi x$. (b) $\hat{\boldsymbol{y}} = \Phi \boldsymbol{x}$. (c) $\hat{\boldsymbol{y}} = \boldsymbol{\Phi} x$.

Fig. 2.1 – Comparaisons des modèles univarié/monocanal (a), multicanal (b) et multivarié (c). Dans (c), $\hat{\times}$ est considéré comme un produit élément par élément selon la dimension M.

Dans le modèle multivarié, $\boldsymbol{\Phi} x$ est considéré comme un produit élément par élément [2] selon la dimension M. La différence entre ces deux modèles sera détaillée dans le prochain paragraphe.

Nous précisons ici que nous travaillons avec des données multicomposantes et non multidimensionnelles. Un signal temporel $\boldsymbol{y} \in \mathbb{R}^{N \times V}$ avec V composantes est abusivement qualifié de multidimensionnel. En fait, ce signal est multicomposant (multicanal ou multivarié selon le modèle d'analyse choisi) et non multidimensionnel. Un signal multidimensionnel (dimension V) serait $\boldsymbol{y} \in \mathbb{R}^{N_1 \times \ldots \times N_v \times \ldots \times N_V}$.

2.1.1 Présentation comparative des modèles

L'approximation parcimonieuse multicanale est connue en anglais sous plusieurs noms : *multichannel sparse approximation* [Gri02, Gri03, LKG06, GRSV07, ER10], *simultaneous sparse approximation* (SSA) [LT03b, LT03a, LT05, TGS06, Tro06], *sparse approximation for multiple measurement vector* (MMV) [CREKD05, CH06], *joint sparsity* [BDS+05] et *multiple sparse approximation* [Rak11]. Dans ce modèle très étudié, toutes les composantes sont décomposées sur le même atome univarié $\phi_m \in \mathbb{R}^N$ et chaque composante v a son propre coefficient $\boldsymbol{x}_m[v] \in \mathbb{R}$, comme illustré en Fig. 2.2(a). Cela signifie que toutes les composantes sont décrites par le même profil, mais avec une énergie différente : l'atome est linéairement mélangé dans les différents canaux :

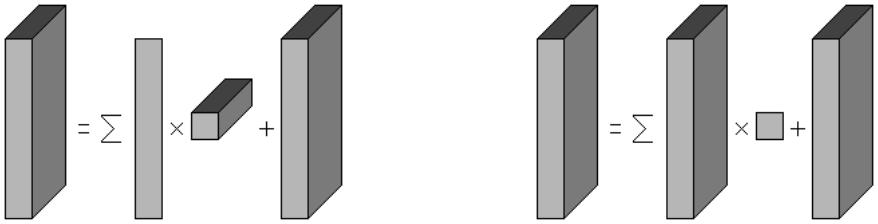
$$\boldsymbol{y} = \sum_{m=1}^{M} \phi_m \boldsymbol{x_m} + \boldsymbol{\epsilon}\,. \tag{2.1}$$

Etant donné que l'atome est univarié, ce modèle est facilement utilisable avec des dictionnaires génériques (cf. Section 1.3). Concernant l'étape de sélection, le *Multichannel* MP original proposé par Gribonval sélectionne l'énergie inter-canaux maximale [Gri03] :

$$m^k = \arg\max_m \sum_{v=1}^{V} \left| \left\langle \boldsymbol{\epsilon}^{k-1}[v], \phi_m \right\rangle \right|^s . \tag{2.2}$$

Cette sélection revient donc à maximiser une norme ℓ_s pour $s = 1$, 2 ou $+\infty$ [CH06]. La recherche par rapport à m du maximum des normes ℓ_s est équivalente à appliquer une norme mixte sur la matrice des produits scalaires $G = \Phi^T \boldsymbol{\epsilon}^{k-1}$: cela engendre une parcimonie structurée qui est semblable au *Group*-LASSO, comme expliqué par Rakotomamonjy [Rak11]. Notons que le *Multichannel* MP [DMMM+05]

2. Il s'agit en réalité d'un produit tensoriel, mais nous n'utiliserons pas ce formalisme par souci de simplicité.



(a) $\boldsymbol{y} = \sum_m \phi_m \boldsymbol{x}_m + \boldsymbol{\epsilon}$ (b) $\boldsymbol{y} = \sum_m \boldsymbol{\phi}_m x_m + \boldsymbol{\epsilon}$

Fig. 2.2 – Illustration des différences entre le modèle multicanal (a) et le modèle multivarié (b).

sélectionne le produit scalaire multicanal maximal :

$$m^k = \arg\max_m \left| \sum_{v=1}^{V} \left\langle \boldsymbol{\epsilon}^{k-1}[v], \phi_m \right\rangle \right| . \tag{2.3}$$

Le modèle multivarié peut être considéré comme l'inverse du modèle multicanal : chaque composante du signal correspond à une composante de l'atome multivarié $\boldsymbol{\phi}_m \in \mathbb{R}^{N \times V}$, et toutes les composantes ont le même coefficient $x_m \in \mathbb{R}$, comme illustré en Fig. 2.2(b). Ainsi, le signal multivarié $\boldsymbol{y}$ est décomposé comme la somme d'atomes multivariés $\boldsymbol{\phi}_m$ tel que :

$$\boldsymbol{y} = \sum_{m=1}^{M} \boldsymbol{\phi}_m x_m + \boldsymbol{\epsilon} . \tag{2.4}$$

Contrairement au modèle multicanal qui impose le fait que la matrice $\phi_m \times \boldsymbol{x}_m$ soit de rang 1, le modèle multivarié est beaucoup plus flexible : il n'y a aucune contrainte sur le rang du sous-espace engendré par l'atome multivarié $\boldsymbol{\phi}_m$. Ainsi, des données provenant de grandeurs physiques différentes, qui ont des profils caractéristiques variés (voire orthogonaux), peuvent être agrégées dans les composantes de l'atome multivarié[3]. Ce modèle a seulement été évoqué dans [GN05] à propos d'un algorithme MP utilisant un modèle de dictionnaire particulier : le dictionnaire multicomposante utilisé est le produit d'un dictionnaire monocomposante avec une matrice de mélange.

Dans la suite, le dictionnaire multivarié $\boldsymbol{\Phi}$ sera considéré comme normé, *i.e.* chaque atome multivarié sera normé tel que $\|\boldsymbol{\phi}_m\|_F = 1$. De plus, il est difficile de définir un atome multivarié à partir de dictionnaires génériques. Le modèle multivarié est donc naturellement tourné vers l'apprentissage de dictionnaire, composé d'atomes multivariés non-paramétriques.

2.1.2 Apprentissages de dictionnaires multicomposantes

Dans ce paragraphe, nous considérons des DLAs qui traitent des signaux multicomposantes avec des dictionnaires multicomposantes [Mal09, Chap. 12]. Remarquons que le modèle employé est différent de ceux décrits précédemment. Dans ce cas, chaque composante $\boldsymbol{y}[v] \in \mathbb{R}^N$ est associée à un atome unicomposante $\boldsymbol{\phi}_m[v] \in \mathbb{R}^N$ multiplié par un coefficient $\boldsymbol{x}_m[v] \in \mathbb{R}$, contrairement au modèle multivarié où chaque atome multivarié $\boldsymbol{\phi}_m$ est multiplié par un coefficient x_m. Pour chaque composante v, nous

3. Elles doivent être seulement homogènes au niveau de leurs dimensions.

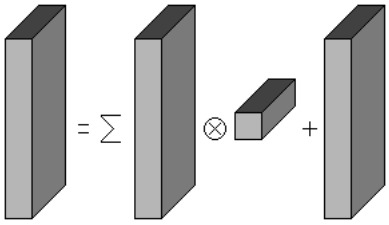


Fig. 2.3 – Illustration du modèle multicomposante : $\boldsymbol{y} = \sum_m \boldsymbol{\phi}_m \otimes \boldsymbol{x}_m + \boldsymbol{\epsilon}$, avec $\otimes$ défini comme un produit composante à composante selon la dimension V.

avons donc :

$$\boldsymbol{y}[v] = \sum_{m=1}^{M} \boldsymbol{\phi}_m[v]\boldsymbol{x}_m[v] + \boldsymbol{\epsilon}[v] \,. \tag{2.5}$$

Ainsi, dans ce modèle, si les atomes multicomposantes ont la possibilité d'être spécifiques dans chaque composante, le signal est transformé en signaux unicomposantes parallèles. Ce modèle est utilisé pour traiter des signaux audio-visuels [MJV+07, MVS09], des signaux électrocardiographiques [MGB+09] et électroencéphalographiques [BPTC09], des images couleur [MES08], des images hyperspectrales [MBSF09] ou encore des images stéréoscopiques [TF11b].

Nous pouvons reformuler le modèle précédent pour y grouper les V composantes :

$$\boldsymbol{y} = \sum_{m=1}^{M} \boldsymbol{\phi}_m \otimes \boldsymbol{x}_m + \boldsymbol{\epsilon} \,, \tag{2.6}$$

avec $\otimes$ défini comme un produit composante à composante selon la dimension V, comme illustré en Fig. 2.3. Les composantes sont seulement liées entre elles lors de l'étape de sélection de l'approximation parcimonieuse. En effet, l'indice sélectionné m^k est choisi conjointement entre les différentes composantes grâce à une norme mixte :

$$m^k = \arg\max_m \sum_{v=1}^{V} \left| \left\langle \boldsymbol{\epsilon}^{k-1}[v], \boldsymbol{\phi}_m[v] \right\rangle \right|^s \,. \tag{2.7}$$

Cette sélection étant proche de l'Eq. (2.2), les algorithmes multicanaux sont utilisés pour ce modèle.

2.2 OMP multivarié

Dans cette section, l'algorithme d'approximation parcimonieuse OMP est étendu au modèle multivarié sous le nom *Multivariate*-OMP (M-OMP). Nous avons choisi d'étendre l'OMP car c'est un algorithme rapide et efficace pour l'apprentissage de dictionnaires invariants par translation. Après avoir présenté le M-OMP, nous évoquerons le lien qui existe entre les modèles univarié et multivarié.

2.2.1 Description du *Multivariate*-OMP

Le problème d'approximation parcimonieuse (1.6) est étendu au modèle multivarié :

$$\min_x \; \| \boldsymbol{y} - \boldsymbol{\Phi} x \|_F^2 \;\text{ t.q. }\; \|x\|_0 \leq K \,. \tag{2.8}$$

Nous présentons maintenant le M-OMP qui résout ce problème d'approximation parcimonieuse multivariée avec le modèle décrit précédemment. A l'itération courante k, le M-OMP sélectionne l'atome qui produit la plus forte décroissance (en valeur absolue) de l'erreur quadratique moyenne $\left\|\boldsymbol{\epsilon}^{k-1}\right\|_F^2$. En notant $\boldsymbol{\epsilon}^{k-1} = x_m \boldsymbol{\phi}_m + \boldsymbol{\epsilon}^k$, nous avons :

$$\frac{\partial \left\| \boldsymbol{\epsilon}^{k-1} \right\|_F^2}{\partial x_m} = 2 \operatorname{Tr}(\boldsymbol{\phi}_m{}^T \boldsymbol{\epsilon}^{k-1}) = 2 \left\langle \boldsymbol{\epsilon}^{k-1}, \boldsymbol{\phi}_m \right\rangle , \tag{2.9}$$

en utilisant les règles de dérivation de la trace d'une matrice [PP08]. L'étape de sélection du M-OMP est donc :

$$m^k = \arg\max_m \left| \left\langle \boldsymbol{\epsilon}^{k-1}, \boldsymbol{\phi}_m \right\rangle \right| \tag{2.10}$$

$$= \arg\max_m \left| \sum_{v=1}^{V} \left\langle \boldsymbol{\epsilon}^{k-1}[v], \boldsymbol{\phi}_m[v] \right\rangle \right| . \tag{2.11}$$

La sélection repose sur la somme des produits scalaires des V composantes, ce qui donne le produit scalaire multivarié, contrairement à la sélection (2.3) où le dictionnaire est univarié. En plus d'une différence sur la morphologie du dictionnaire utilisé, les modèles multicanal et multivarié ne sélectionnent pas les mêmes atomes. Pour mieux illustrer cela, nous nous plaçons dans un cas complexe. A cause des modules (valeurs absolues), la colinéarité ou la non-colinéarité des V composantes $G_m[v] = \left\langle \boldsymbol{\epsilon}^{k-1}[v], \phi_m \right\rangle$ ne sont pas prises en compte dans l'Eq. (2.2), et la sélection multicanale se fait seulement sur l'énergie. La sélection multivariée de l'Eq. (2.11) est plus exigente, et cherche le compromis optimal parmi les V composantes $G_m[v] = \left\langle \boldsymbol{\epsilon}^{k-1}[v], \boldsymbol{\phi}_m[v] \right\rangle$. Elle retient l'atome qui représente au mieux le résidu dans chaque composante. Les sélections sont équivalentes si tous les $G_m[v]$ sont colinéaires (et de mêmes signes). Les différences entre ces deux types de sélection ont été aussi discutées dans [GN05, DMMM+05].

Par la suite, la description est donnée dans le cas d'invariance par translation dans l'Algorithme 6, et nous faisons référence à la description de l'OMP décrite dans l'Algorithme 3 (Section 1.2.2). Le problème (2.8) devient :

$$\min_x \left\| \boldsymbol{y}(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau} \boldsymbol{\psi}_l(t-\tau) \right\|_F^2 \text{ t.q. } \|x\|_0 \leq K . \tag{2.12}$$

La sélection cherche la corrélation Γ maximale entre le résidu et les noyaux :

$$(l^k, \tau^k) = \arg\max_{l,\tau} \left| \sum_{v=1}^{V} \Gamma \left\{ \boldsymbol{\epsilon}^{k-1}[v], \boldsymbol{\psi}_l[v] \right\} (\tau) \right| . \tag{2.13}$$

Le dictionnaire actif $\boldsymbol{D}^k$ est aussi multivarié (étape 7). Pour la projection orthogonale (étape 8), le signal multivarié $\boldsymbol{y} \in \mathbb{R}^{N \times V}$ (*resp.* dictionnaire $\boldsymbol{D}^k \in \mathbb{R}^{N \times k \times V}$) est déplié verticalement le long de la dimension des composantes dans un vecteur unicomposante $\boldsymbol{y}\rfloor \in \mathbb{R}^{NV \times 1}$ (*resp.* matrice $\boldsymbol{D}^k\rfloor \in \mathbb{R}^{NV \times k}$). Ensuite, la projection orthogonale de $\boldsymbol{y}\rfloor$ sur $\boldsymbol{D}^k\rfloor$ définie par :

$$x^k = \arg\min_x \left\| \boldsymbol{y}\rfloor - \boldsymbol{D}^k\rfloor x \right\|_2^2 \tag{2.14}$$

est calculée récursivement, comme dans le cas unicomposante, en utilisant l'inversion matricielle par bloc [PRK93]. A la fin de l'algorithme, le M-OMP fournit une approximation K-parcimonieuse multivariée

de $\boldsymbol{y}$:

$$\hat{\boldsymbol{y}}^K(t) = \sum_{k=1}^{K} x_{l^k,\tau^k}\, \boldsymbol{\psi}_{l^k}(t-\tau^k)\,. \tag{2.15}$$

Comparée à l'OMP, la complexité du M-OMP (déterminée par l'étape 4) est seulement multiplié par un facteur V, le nombre de composantes.

Algorithm 6 : $x = \text{Multivariate\_OMP}\ (\boldsymbol{y}, \boldsymbol{\Psi})$

1: **initialisation** : $k=1,\ \boldsymbol{\epsilon}^0 = \boldsymbol{y},$ dictionnaire $\boldsymbol{D}^0 = \varnothing$
2: **repeat**
3: **for** $l \leftarrow 1, L$ **do**
4: Corrélations : $C_l^k(\tau) \leftarrow \sum_{v=1}^{V} \Gamma\left\{\boldsymbol{\epsilon}^{k-1}[v], \boldsymbol{\psi}_l[v]\right\}(\tau)$
5: **end for**
6: Sélection : $(l^k, \tau^k) \leftarrow \arg\max_{l,\tau} \left| C_l^k(\tau) \right|$
7: Dictionnaire actif : $\boldsymbol{D}^k \leftarrow \left[\, \boldsymbol{D}^{k-1}, \boldsymbol{\psi}_{l^k}(t-\tau^k) \,\right]$
8: Coefficients actifs : $x^k \leftarrow \arg\min_x \left\| \boldsymbol{y}\rfloor - \boldsymbol{D}^k\rfloor x \right\|_2^2$
9: Résidu : $\boldsymbol{\epsilon}^k \leftarrow \boldsymbol{y} - \boldsymbol{D}^k x^k$
10: $k \leftarrow k+1$
11: **until** critère d'arrêt

2.2.2 Remarques sur le modèle multivarié

Pour un signal multicomposante $\boldsymbol{y} \in \mathbb{R}^{N\times V}$ de nature non temporelle, *i.e.* avec $N=1$, l'approche multivariée est traitée en utilisant des signaux vectorisés. Le signal multicomposante $\boldsymbol{y} \in \mathbb{R}^{N\times V}$ est vectorisé verticalement en $\boldsymbol{y}\rfloor \in \mathbb{R}^{NV\times 1}$, et le dictionnaire $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M\times V}$ en $\boldsymbol{\Phi}\rfloor \in \mathbb{R}^{NV\times M}$. Après avoir appliqué un OMP classique (cf. Section 1.2.2) sur ces variables vectorisées, le traitement est équivalent à celui produit par l'algorithme M-OMP. En effet, le modèle linéaire multivarié

$$\boldsymbol{y} = \sum_{m=1}^{M} x_m \boldsymbol{\phi}_m + \boldsymbol{\epsilon} \quad \text{se vectorise en} \quad \boldsymbol{y}\rfloor = \sum_{m=1}^{M} x_m \boldsymbol{\phi}_m\rfloor + \boldsymbol{\epsilon}\rfloor\,. \tag{2.16}$$

Plusieurs raisons soutiennent l'introduction d'un tel modèle multivarié.

D'une part, dans le cas de données temps-séries, les différentes composantes sont acquises simultanément, et le modèle multivarié permet de conserver cette structure temporelle des données. De plus, ces composantes peuvent être des grandeurs physiques très hétérogènes. Vectoriser les composantes engendre une perte du sens physique des signaux et des noyaux du dictionnaire, et plus particulièrement quand les composantes ont des profils très différents. Nous préférons considérer les données temps-séries étudiées comme étant multicomposantes : les signaux et les atomes du dictionnaire possèdent plusieurs composantes simultanées, comme illustré dans les expérimentations suivantes.
Pour l'implémentation algorithmique, dans le cas d'invariance par translation, le modèle multivarié est plus facile à implémenter et possède une complexité moins grande que le modèle classique avec les données vectorisées (cf. Annexe 7.2).

D'autre part, le modèle multivarié a vocation à être généralisé. Pour notre propos, nous considérons le signal multivarié $\boldsymbol{y} \in \mathbb{R}^{N\times V}$, les atomes multivariés $\boldsymbol{\phi}_m \in \mathbb{R}^{N\times U_m}$ ayant chacun leur dimension spécifique

U_m, et nous introduisons des matrices quelconques $R_m \in \mathbb{R}^{U_m \times V}$. Le modèle multivarié peut ainsi se généraliser :

$$\boldsymbol{y} = \sum_{m=1}^{M} x_m \boldsymbol{\phi}_m R_m + \boldsymbol{\epsilon} \,. \tag{2.17}$$

Les matrices R_m empêchent désormais la norme de Frobenius de se vectoriser en une norme ℓ_2. En outre, d'autres métriques [CBA13] peuvent être utilisées dans le critère d'optimisation, résolu par de nouveaux algorithmes. Dans la suite de cette thèse, nous étudierons les cas de matrices de rotation, où $\forall m \in \mathbb{N}_M, U_m = V$, pour $V = 2$ (cf. Chapitre 4), $V = 3$ (cf. Chapitre 5) et $V = n$ quelconque (cf. Chapitre 5). Le modèle étant présenté de cette façon, les méthodes introduites ultérieurement pourront être vues très simplement comme des spécifications de ce modèle multivarié. Les composantes acquises simultanément, mais qui sont actuellement considérées comme indépendantes, seront mélangées par des rotations 2D, 3D et nD. Ceci est plus intuitif pour des atomes multivariés contrairement à des atomes vectorisés.

Ainsi, le modèle multivarié est introduit ici pour plus de clarté, pour des raisons algorithmiques et dans un souci de future généralisation.

2.3 DLA multivarié

Dans cette section, nous présentons un algorithme d'apprentissage de dictionnaire adapté au modèle multivarié et appelé *Multivariate*-DLA (M-DLA).

2.3.1 Description du *Multivariate*-DLA

Nous présentons maintenant le *Multivariate* DLA. Nous disposons d'un ensemble de signaux d'entraînement $\boldsymbol{Y} = \{\boldsymbol{y}_p\}_{p=1}^{P}$. Notre algorithme M-DLA (Algorithme 7) est de nature *online* (cf. Section 1.3.2). Il s'agit donc d'une alternance entre une approximation parcimonieuse multivariée et la mise à jour du dictionnaire multivarié. L'approximation parcimonieuse multivariée de chaque signal $\boldsymbol{y}_p$ est réalisée par le M-OMP (étape 4) :

$$x_p = \arg\min_x \left\| \boldsymbol{y}_p(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau} \boldsymbol{\psi}_l(t-\tau) \right\|_F^2 \quad \text{t.q.} \quad \|x\|_0 \leq K \,, \tag{2.18}$$

suivie de la mise à jour du dictionnaire (étape 5) :

$$\boldsymbol{\Psi} = \arg\min_{\boldsymbol{\Psi}} \left\| \boldsymbol{y}_p(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau;p} \boldsymbol{\psi}_l(t-\tau) \right\|_F^2 \quad \text{t.q.} \quad \forall l \in \mathbb{N}_L, \|\boldsymbol{\psi}_l\|_F = 1 \,. \tag{2.19}$$

Pour effectuer l'optimisation de ce critère, nous utilisons une descente de gradient de 2$^{\text{nd}}$ ordre de Levenberg-Marquardt stochastique [MNT04]. Ce choix augmente la vitesse de convergence, en mélangeant les méthodes de gradient stochastique et de Gauss-Newton. Le critère est dérivé en utilisant les règles de dérivation de [PP08] :

$$-\frac{\partial \| \boldsymbol{\epsilon}_p(t) \|_F^2}{\partial \psi_l} = 2 \sum_{\tau \in \sigma_l} x_{l,\tau;p} \, \epsilon_\tau(t) \,, \tag{2.20}$$

avec $\underline{\boldsymbol{\epsilon}}_\tau$ représentant l'erreur localisée au décalage τ et restreinte au support temporel de $\boldsymbol{\psi}_l$, *i.e.* $\underline{\boldsymbol{\epsilon}}_\tau = \boldsymbol{\epsilon}|_{t=\tau..\tau+T_l}$, avec T_l la longueur de $\boldsymbol{\psi}_l$. Nous appelons i l'itération courante du M-DLA. Pour chaque noyau multivarié $\boldsymbol{\psi}_l$, la mise à jour est donnée par :

$$\boldsymbol{\psi}_l^i(\underline{t}) = \boldsymbol{\psi}_l^{i-1}(\underline{t}) + (H_l^i + \lambda^i.I)^{-1} \cdot \sum_{\tau \in \sigma_l} x_{l,\tau;p}^i \, \boldsymbol{\epsilon}_p^{i-1}(\underline{t}+\tau) \,, \tag{2.21}$$

avec $\underline{t}$ pour les indices d'échantillons limités au support temporel de $\boldsymbol{\psi}_l$, λ le pas de descente adaptatif, et H_l le Hessien approché comme indiqué en Annexe 7.3. Cette étape est appelé Mise_à_jour_LM (étape 5). Le modèle multivarié est pris en compte dans la mise à jour du dictionnaire, avec toutes les composantes $\boldsymbol{\psi}_l[v]$ du noyau multivarié $\boldsymbol{\psi}_l$ mises à jour simultanément. De plus, les noyaux sont normés à la fin de chaque mise à jour, et leurs longueurs peuvent être modifiées.

Au début de l'algorithme, l'initialisation des noyaux (étape 1) se fait sur du bruit blanc Gaussien. A la fin, différents critères d'arrêt peuvent être utilisés (étape 8) : un seuillage sur le rRMSE calculé sur l'ensemble d'apprentissage, ou un seuil sur i, le nombre d'itérations. Dans le M-DLA, le M-OMP est arrêté par seuillage sur le nombre d'itérations. Le rRMSE ne peut pas être utilisé puisque sur les premières itérations, les noyaux de bruit blanc ne peuvent pas engendrer un pourcentage donné de l'espace étudié.

Algorithm 7 : Ψ = Multivariate_DLA $(\{\boldsymbol{y}_p\}_{p=1}^P)$

1: **initialisation** : $i = 1, \boldsymbol{\Psi}^0 = \{L$ noyaux de bruit blanc$\}$
2: **repeat**
3: **for** $p \leftarrow 1, P$ **do**
4: Approximation parcimonieuse : $x_p^i \leftarrow$ M-OMP $(\boldsymbol{y}_p, \boldsymbol{\Psi}^{i-1})$
5: Mise à jour du dictionnaire : $\boldsymbol{\Psi}^i \leftarrow$ Mise_à_jour_LM $(\boldsymbol{y}_p, x_p^i, \boldsymbol{\Psi}^{i-1})$
6: $i \leftarrow i + 1$
7: **end for**
8: **until** critère d'arrêt

2.3.2 Remarques sur le *Multivariate*-DLA

Notre algorithme d'apprentissage est de nature *online*, et nous pouvons tolérer des fluctuations dans la MSE. L'aspect stochastique de l'optimisation *online* permet de ressortir d'un minimum local. Les optimisations non-convexes du M-OMP et du M-DLA et la nature stochastique de l'apprentissage *online* ne permettent pas de garantir la convergence du M-DLA vers le minimum global. Cependant, nous trouvons un dictionnaire, minimum local ou global, qui assure la parcimonie des décompositions des signaux d'intérêt.

Concernant le réglage des paramètres, il y a de nombreuses stratégies pour le pas de descente adaptatif : nous choisissons la procédure classique $\lambda^i = \lambda_0 \cdot i$ avec $\lambda_0 = 1$). Lors de la mise à jour, la longueur des noyaux est modifiée selon une heuristique. Les noyaux sont allongés s'il y a de l'énergie sur les bords et raccourcis sinon, de telle sorte que les noyaux soient à bords nuls, ce qui évite les discontinuités dans les décompositions réalisées avec ce dictionnaire. Nous calculons la norme sur T_{l_bord} échantillons des deux bords du noyau l, et nous les divisons par la norme du noyau complet. Si ce ratio est supérieur à un certain seuil, nous allongeons le noyau de $T_{l\_bord}/2$ échantillons, si le ratio est inférieur à un certain

autre seuil, alors nous le raccourcissons. Le rapport 2 entre la longueur d'estimation et la longueur allongée/raccourcie évite les instabilités du processus. D'autres heuristiques peuvent être mises en place, comme dans [SP11] par exemple.

2.4 Comparaison des méthodes sur des données simulées

Dans cette section, nous comparons notre méthode à d'autres DLAs. Comme les avantages de l'apprentissage *online* ont déjà été mis en avant dans [AE08, MBPS10, SE10], nous tournons notre expérience vers la robustesse à la translation temporelle. Le M-DLA est utilisé dans un cas monocomposante pour être comparé avec les autres méthodes existantes : la K-SVD [AEB06], la version invariante par translation de la K-SVD [Aha06] appelée SI-K-SVD, et le ILS-DLA invariant par translation [SHA06] (le facteur de décalage est réglé à 1), qui sera appelé SI-ILS-DLA par la suite.

La comparaison est basée sur l'expérience décrite dans [Aha06, Chap. 6]. Un dictionnaire Ψ de $L = 45$ noyaux est créé aléatoirement sur du bruit blanc uniforme et la longueur des noyaux est de $T = 18$ échantillons. L'ensemble d'apprentissage est composé de $P = 2000$ signaux de longueur $N = 20$, et il est généré à partir du dictionnaire. Pour les noyaux, les décalages circulaires ne sont pas autorisés, et donc seulement trois décalages sont possibles. Chaque signal d'entraînement est composé de la somme de trois atomes, pour lesquels les amplitudes, les indices de noyaux et les paramètres de décalages sont tirés aléatoirement. Un bruit blanc Gaussien est aussi ajouté à plusieurs RSB : 10, 20 et 30 dB, et sans bruit. Tous les algorithmes sont appliqués avec le même ensemble d'apprentissage généré ainsi, avec l'initialisation du dictionnaire réalisée avec des signaux d'entraînement, et avec l'étape d'approximation parcimonieuse réalisée par OMP. Les dictionnaires appris $\hat{\Psi}$ sont obtenus après 80 itérations. La K-SVD basique est aussi testée, avec l'espoir de retrouver un dictionnaire de 135 atomes (les 45 noyaux décalés aux trois positions possibles).

Dans l'expérience, un noyau appris est considéré comme détecté, *i.e.* retrouvé, si son produit scalaire c_l avec son noyau original correspondant est tel que :

$$c_l = \left| \left\langle \psi_l, \hat{\psi}_l \right\rangle \right| \geq 0.99 \,. \tag{2.22}$$

Le seuil élevé de 0.99 a été choisi dans [Aha06]. Pour chaque algorithme d'apprentissage, le taux de détection de noyaux (moyenné sur 5 tests) est tracé en fonction du niveau de bruit (Fig. 2.4).

Cette expérience teste seulement la précision de l'algorithme. Dans notre cas, l'apprentissage *online* donne un apprentissage rapide mais pas aussi précis à cause du bruit stochastique induit par le choix aléatoire des signaux à chaque itération. Nous observons que 80% des $\{c_l\}_{l=1}^{L}$ sont entre 0.97 et 1.00, avec seulement quelques-uns au-dessus du seuil sévère de 0.99. Pour être comparable aux algorithmes *batch*, qui sont plus précis à chaque pas, la stratégie classique pour le pas adaptatif (proposée en Section 2.3) est adaptée aux contraintes de cette expérience. Avec 2000 signaux d'entraînement, nous préférons garder un pas constant pour une boucle sur l'ensemble d'apprentissage. De plus, le pas est incrémenté plus rapidement pour atteindre une convergence satisfaisante après 80 itérations. Pour les 40 premières itérations, le pas est défini comme : $\lambda^i = (i - p + 1)^{1.5}$, et il est ensuite gardé constant pour les dernières itérations : $\lambda^i = 40^{1.5}$. Les résultats obtenus sont tracés en Fig. 2.4.

Sur la Fig. 2.4, les résultats de la K-SVD comparés à ceux des autres DLAs montrent la pertinence d'avoir un modèle invariant par translation pour traiter de telles données. Pour les DLAs inva-

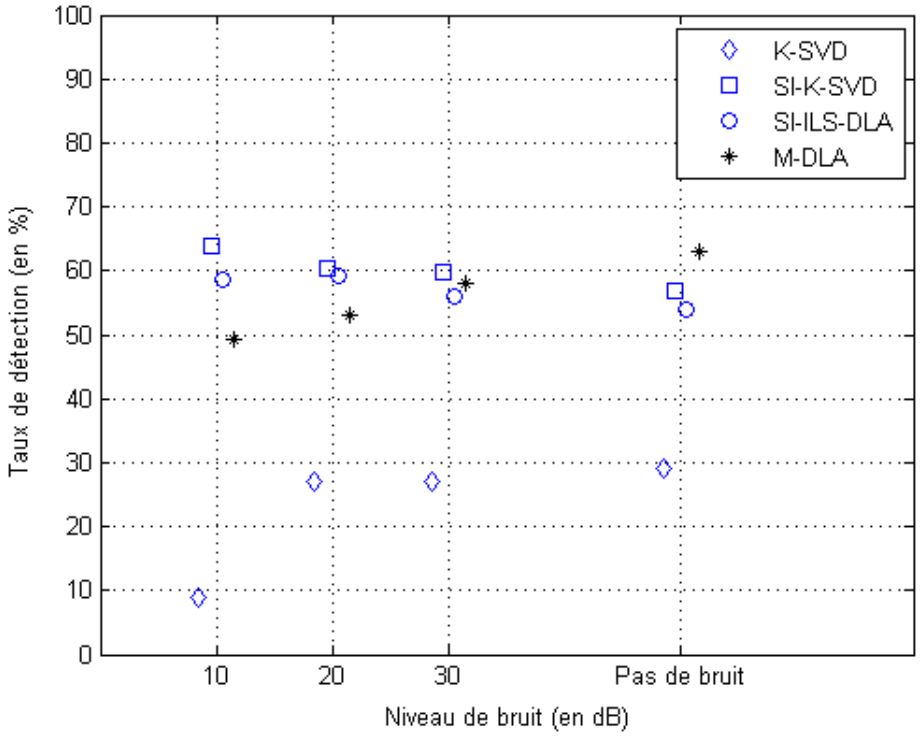


Fig. 2.4 – Taux de détection en fonction du niveau de bruit, pour la K-SVD [AEB06] (diamants), SI-K-SVD [Aha06] (carrés), SI-ILS-DLA [SHA06] (cercles) et M-DLA (étoiles).

riants par translation, les résultats soulignent leur capacité à retrouver des motifs sous-jacents invariants par translation. Cependant, nous observons que les performances du M-DLA décroissent avec le bruit, contrairement à la SI-K-SVD et au SI-ILS-DLA, qui semblent ne pas être affectés. Malgré sa mise à jour stochastique, notre algorithme retrouve le dictionnaire original dans des proportions similaires aux approches *batch*. Cette expérience corrobore l'analyse de [BB08] concernant l'apprentissage, où chaque étape n'a pas besoin d'être minimisée exactement pour converger vers la solution espérée.

Nous avons testé notre méthode sur des données simulées afin de la comparer aux méthodes existantes, dans un cadre univarié. Dans la section suivante, nous allons appliquer notre méthode à des données réelles multivariées.

2.5 Applications aux signaux électroencéphalographiques

Dans cette section, nous allons brièvement expliquer les signaux électroencéphalographiques (EEG) et les outils utilisés pour les représenter. Nous allons ensuite réaliser deux expériences pour mettre en évidence l'apport des dictionnaires multivariés à ce domaine.

2.5.1 Etat de l'art de l'analyse temps-fréquence pour les EEG

L'électroencéphalographie mesure les activités électriques produites par les potentiels post-synaptiques de groupes de neurones. L'acquisition est réalisée à l'aide d'un casque muni de plusieurs électrodes. Ces signaux multi-électrodes ont la particularité d'être très bruités. L'EEG est très utilisée par les neurophy-

siologistes dans le diagnostic de certaines maladies [NLdS04]. L'observation des potentiels évoqués, de certaines activités oscillatoires dans des bandes de fréquence spécifiques, ou encore de certaines activités transitoires permet en effet de déceler certains dysfonctionnements cérébraux.

En analyse temps-fréquence, des dictionnaires classiques comme Gabor ou ondelettes ont été utilisés [SC07, Dur07] mais sont limités pour représenter une telle diversité de motifs ; en outre, ils introduisent un *a priori* lors de l'analyse à cause de la forme des atomes utilisés. Représenter efficacement de tels signaux est donc un défi. Nous nous proposons d'appliquer l'apprentissage de dictionnaire sur ces données afin d'améliorer la représentation et la compréhension des activités EEG.

Dans les paragraphes suivants, nous faisons brièvement la revue des différents outils d'analyse temps-fréquence appliqués aux données EEG.

Décompositions multicanales

Au début, l'algorithme MP (cf. Section 1.2.2) était utilisé sur des signaux monocanaux $y \in \mathbb{R}^N$ [DB01], *i.e.* en prenant en compte seulement une électrode. Le dictionnaire classiquement utilisé en analyse EEG est le dictionnaire de Gabor. Ses atomes génériques ϕ_{Gabor} sont paramétrés en temps-fréquence-échelle tels que [Mal09] :

$$\phi_{\text{Gabor}}(t) = \frac{1}{\sqrt{s}} \cdot g\left(\frac{t-\alpha\tau}{s}\right) \cdot \cos\left(\, 2\pi f t + \varphi \,\right) \,, \tag{2.23}$$

où $g(t) = \beta \cdot e^{-\pi t^2}$ est une fenêtre Gaussienne, β le facteur de normalisation, s l'échelle, τ le décalage temporel, α le facteur de décalage, f la fréquence et parfois, φ la phase interne de l'atome de Gabor. Remarquons que le dictionnaire de Gabor multi-échelles n'est pas invariant par translation à proprement parler, puisque le facteur de décalage α, qui dépend de l'échelle dyadique s, n'est pas égal à 1 [Mal09, Chap. 12].

L'utilisation du MP a été étendue pour les signaux $\boldsymbol{y} \in \mathbb{R}^{N \times V}$ composés de V électrodes. Le modèle multicanal décrit en Section 2.1 permet de prendre en compte toutes les électrodes. Comme déjà remarqué, l'hypothèse sous-jacente de ce modèle est que les événements EEG sont considérés comme diffusés spatialement dans les différents canaux, formant ainsi une matrice de rang 1. Dans [Dur07], différentes sélections sont énumérées :

- le *Multichannel* MP original [Gri03], appelé MMP_1 par Durka, sélectionne l'énergie maximale comme dans l'Eq. (2.2) ;
- le *Multichannel* MP [DMMM+05] appelé MMP_2 sélectionne le produit scalaire multicanal maximal comme dans l'Eq. (2.3). Les valeurs absolues de la sélection (2.2) permettent à l'atome ϕ_m d'être en phase ou en opposition de phase avec la composante $\boldsymbol{\epsilon}^{k-1}[v]$, contrairement à la sélection plus contraignante (2.3) qui préfère que ϕ_m soit en phase avec $\boldsymbol{\epsilon}^{k-1}[v]$ (donnant ainsi les mêmes polarités à travers les canaux).
- le *Multichannel* MP [MDMM+05] appelé MMP_3 sélectionne l'énergie maximale comme le MMP_1, mais avec des coefficients complexes qui permettent d'avoir des phases variables : chaque canal v possède sa propre phase φ_v, comme aussi étudié dans [GHANZ07] ;
- le *Multichannel* MP [SKM+09b], qui est appelé MMP_4, sélectionne le produit scalaire multicanal maximal comme le MMP_2, avec des coefficients complexes qui permettent d'avoir une phase φ_v dans chaque canal ;

Remarquons que le nom *multivariate* est utilisé dans [SKM+09a] pour désigner le MMP_2 [DMMM+05] appliqué aux données MEG, et dans [SKM+09b] pour son extension complexe MMP_4 ; cependant ils sont complètement différents des méthodes introduites en Sections 2.2 et 2.3.

D'autres algorithmes effectuent des décompositions de signaux EEG comme par exemple, la décomposition multicanale proposée dans [KMLVS01] reposant sur la *method of frames* [Dau88] ou le modèle PARAFAC (*Parallel Factor Analysis*) étendu au cas d'invariance par translation [MHA+08].

Apprentissage de dictionnaire

A notre connaissance, peu d'études ont proposé d'appliquer l'apprentissage de dictionnaire aux données EEG. En 2005, Jost *et al.* appliquent l'algorithme MoTIF [JVLG05], qui est un DLA invariant par translation, à des signaux EEG. Ils apprennent un dictionnaire de noyaux, mais seulement dans un cas monocanal qui ne prend pas en compte l'aspect spatial des données. En 2011, Hamner *et al.* appliquent la K-SVD [HCdRM11] pour effectuer un apprentissage de dictionnaire spatial ou temporel. L'apprentissage spatial est efficace et peut être vu comme une généralisation de l'algorithme des N-Microétats [PMML95]. A l'inverse, les résultats avec l'apprentissage temporel ne sont pas convaincants, principalement parce que le modèle d'invariance par translation n'est pas pris en compte. En 2012, Noirhomme *et al.* appliquent aussi la K-SVD [NSL+12b, NSL+12a], mais dans un cas monocanal et sans invariance par translation.

Comme nous l'avons vu dans le Chapitre 1, l'apprentissage de dictionnaire donne une représentation plus efficace que la PCA ou l'ICA très utilisées en analyse EEG [CGPJ08]. De plus, de telles méthodes fournissent seulement une base (avec $M = N$), et ne sont donc pas adaptées pour faire face à l'invariance par translation réclamée par les données EEG.

Bilan

Pour résumer ce bref état de l'art de l'analyse EEG, différents éléments montrent déjà la pertinence d'utiliser un modèle d'invariance par translation temporelle. C'est la raison pour laquelle la dictionnaire de Gabor, qui est quasi invariant par translation, est bien adapté pour représenter les données EEG avec des *multichannel* MPs [Dur07]. Dans les décompositions multicanales, les différents MPs essayent d'être le plus flexible possible pour représenter au mieux la variabilité spatiale.

L'approche multivariée que nous proposons prend en compte ces deux aspects : un modèle temporel invariant par translation et une flexibilité spatiale qui considère tous les canaux. De plus, seulement deux hypothèses sont faites dans cette approche : le bruit des EEG est additif et Gaussien, et les événements EEG d'intérêt sont statistiquement répétés suite au même stimulus. Il n'y a aucune hypothèse temporelle ou spatiale faite sur le dictionnaire, les résultats sont donc au maximum guidés par les données.

2.5.2 Expérience 1 : apprentissage et décompositions

Réalisées sur des données EEG réelles (issues des BCI Competition II [BS02] et IV [TMA+12]), les deux expériences présentées par la suite ont pour but de montrer que les noyaux multivariés appris sont informatifs (Expérience 1) et interprétables (Expérience 2). Dans cette première expérience, notre méthode est appliquée aux données EEG, et ensuite comparée aux méthodes usuellement utilisées pour

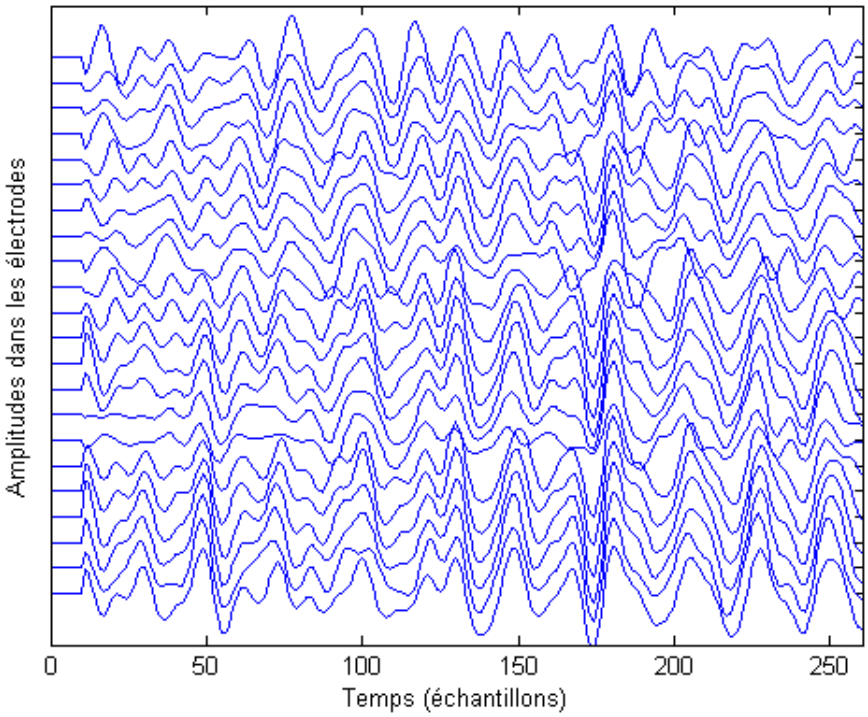


Fig. 2.5 – Signal EEG $\boldsymbol{y}_{p=1}$ échantillonné à 250 Hz avec $V = 22$ électrodes.

analyser les EEG (cf. Section 2.5.1) pour mettre en exergue la nouveauté du modèle multivarié et ses performances représentatives.

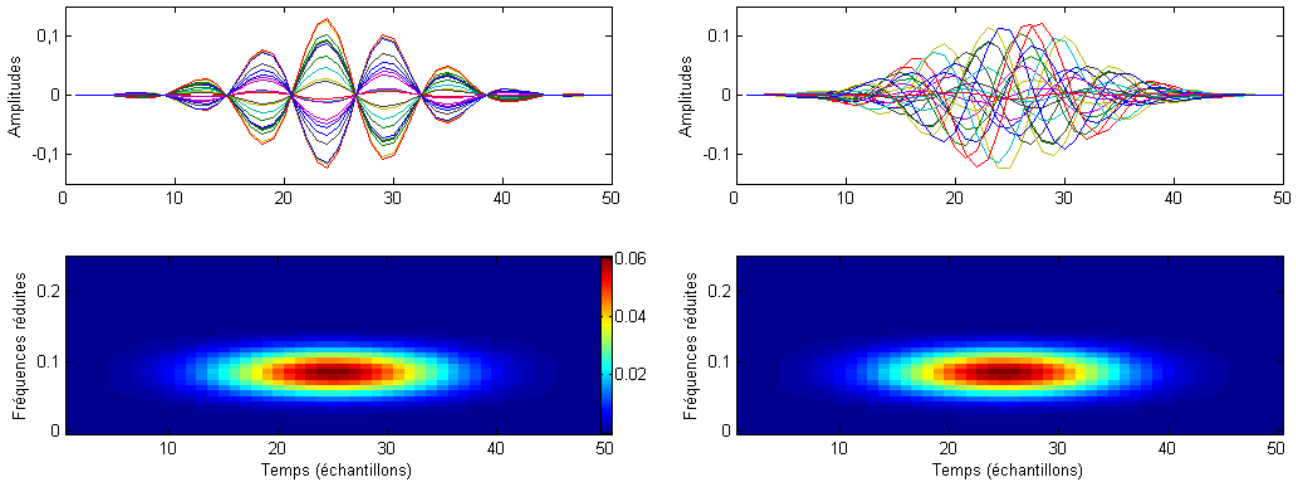
Données EEG

Nous utilisons les signaux d'imagerie motrice des données 2a de BCI Competition IV [TMA+12]. Il y a quatre classes de tâches motrices, mais elles ne sont pas prises en compte dans cette expérience. Les signaux EEG sont échantillonnés à 250 Hz et possèdent $V = 22$ électrodes. Chaque signal est composé de $N = 501$ échantillons temporels, durant lesquels le sujet effectue l'une des quatre tâches. Les données viennent de 9 sujets, et les signaux sont divisés entre un ensemble d'apprentissage et un ensemble de test. Chaque ensemble est composé de $P = 288$ signaux.
Les données brutes sont filtrées entre 8 et 30 Hz (l'imagerie motrice concerne les bandes μ (8 à 13 Hz) et β (13 à 20 Hz)), puis une complétion de zéro est effectuée pour obtenir des signaux de $N = 512$ échantillons. Les données issues de ce prétraitement sont les données d'entrée du M-DLA. Les premiers échantillons du signal EEG $\boldsymbol{y}_{p=1}$ sont affichés en Fig. 2.5.

Modèles et comparaisons

Le M-DLA est appliqué sur l'ensemble d'apprentissage du premier sujet, et un dictionnaire de $L = 20$ noyaux est appris avec 100 itérations [4].

4. Durant l'apprentissage, le contrôle de la longueur des noyaux est délicat à cause du faible RSB des signaux. Pour le M-DLA original, les noyaux sont d'abord initialisés à une longueur arbitraire T^i, et ils sont ensuite allongés ou raccourcis lors des mises à jour, en fonction de l'énergie présente sur les bords. Cependant, avec des données aussi bruitées, les noyaux



(a) Atome de Gabor réel.

(b) Atome de Gabor complexe.

Fig. 2.6 – Exemple d'atomes de Gabor réel (a) et complexe (b), avec les profils temporels de chaque noyau (haut) et la visualisation temps-fréquence (bas). Dans le cas complexe, seule la phase des différents canaux varie, pas le contenu spectral.

Pour mettre en évidence la nouveauté du modèle multivarié, les dictionnaires multicomposantes sont comparés avec $V = 22$ composantes. Sur les Fig. 2.6(a) et 2.6(b), en haut, les amplitudes $x_m \times \phi_m$ (ordonnées) d'un atome sont représentées en fonction des échantillons (abscisses), et en bas, les spectrogrammes en fréquences réduites (ordonnées) sont représentés en fonction des échantillons (abscisses). Un atome de Gabor réel est affiché en Fig. 2.6(a), basé sur le MMP_1 [Gri03] ou le MMP_2 [DMMM+05] qui donne le même genre d'atomes. Les paramètres de l'atome de Gabor ont été choisis aléatoirement. En Fig. 2.6(b) est affiché un atome de Gabor complexe basé sur le MMP_3 [MDMM+05, GHANZ07] ou le MMP_4 [SKM+09b]. Etant donné que cet atome possède une phase spécifique pour chaque canal, il est plus adaptatif que le premier. Cependant, dans ces deux cas, le contenu spectral est identique dans chaque canal.

En Fig. 2.7, deux des noyaux multivariés appris sont affichés, les noyaux $l = 9$ (haut) et $l = 17$ (bas). Les composantes du noyau de la Fig. 2.7(haut) sont similaires, ce qui est adapté pour représenter des données ayant des formes proches sur toutes les composantes, comme le signal $\boldsymbol{y}_1$ aux alentours des échantillons $t = 175$ (cf. Fig. 2.5). Les composantes du noyau de la Fig. 2.7(bas) ne sont pas si différentes : elles apparaissent comme étant continûment déformées, ce qui correspond bien aux données structurées comme le signal $\boldsymbol{y}_1$ autour des échantillons $t = 75$ (cf. Fig. 2.5). De plus, les composantes possèdent des contenus spectraux variés, comme affiché en Fig. 2.8, contrairement aux atomes de Gabor, qui ressemblent plus à des bancs de filtres monocanaux (Fig. 2.6(a)(bas) et 2.6(b)(bas)). Dans le modèle multivarié, chaque composante a son propre profil, et donc son propre contenu spectral, ce qui donne une excellente adaptivité spatiale.

Si les deux noyaux considérés ci-dessus sont de rangs pleins, nous calculons leurs valeurs singulières

ont tendance à s'allonger sans s'arrêter. Un nouveau contrôle est donc mis en place : une taille limite T^m borne les noyaux sur les premiers 2/3 des itérations, et ensuite, la limite est fixée à $T^m + 40$ pour les dernières itérations. Cela permet aux noyaux de commencer à converger, et ensuite d'avoir la possibilité d'obtenir des bords quasi nuls.

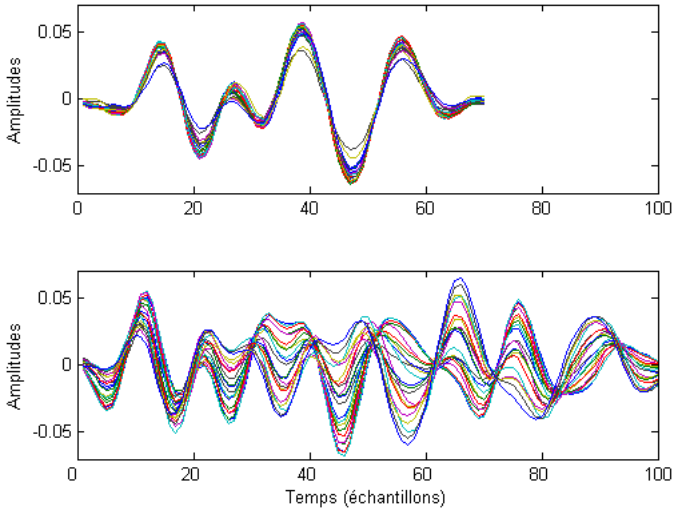


Fig. 2.7 – Noyaux du dictionnaire multivarié appris : le noyau $l = 9$ (haut) et le noyau $l = 17$ (bas).

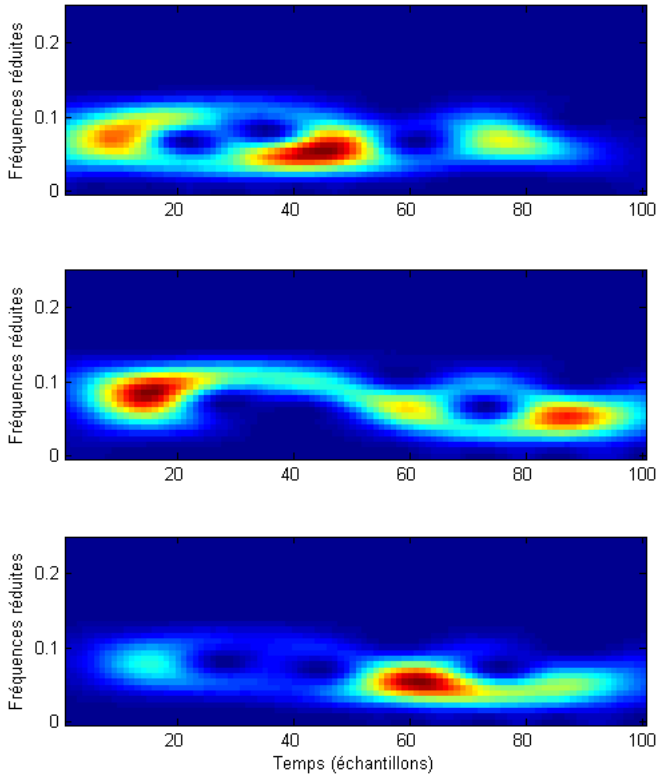


Fig. 2.8 – Spectrogrammes de trois composantes du noyau $l = 17$: composante $v = 10$ (haut), composante $v = 14$ (milieu) et composante $v = 20$ (bas).

par SVD et nous les affichons en Fig. 2.9. Les valeurs sont tracées par ordre décroissant (haut) et en échelle logarithmique (bas) pour le noyau $l = 9$ (gauche) et $l = 17$ (droite). La première valeur singulière du noyau $l = 9$ est très élevée et les autres sont faibles. Il est donc réaliste d'approcher ce noyau multivarié par un noyau multicanal de rang 1. Concernant le noyau $l = 17$, nous observons qu'il y a plusieurs valeurs

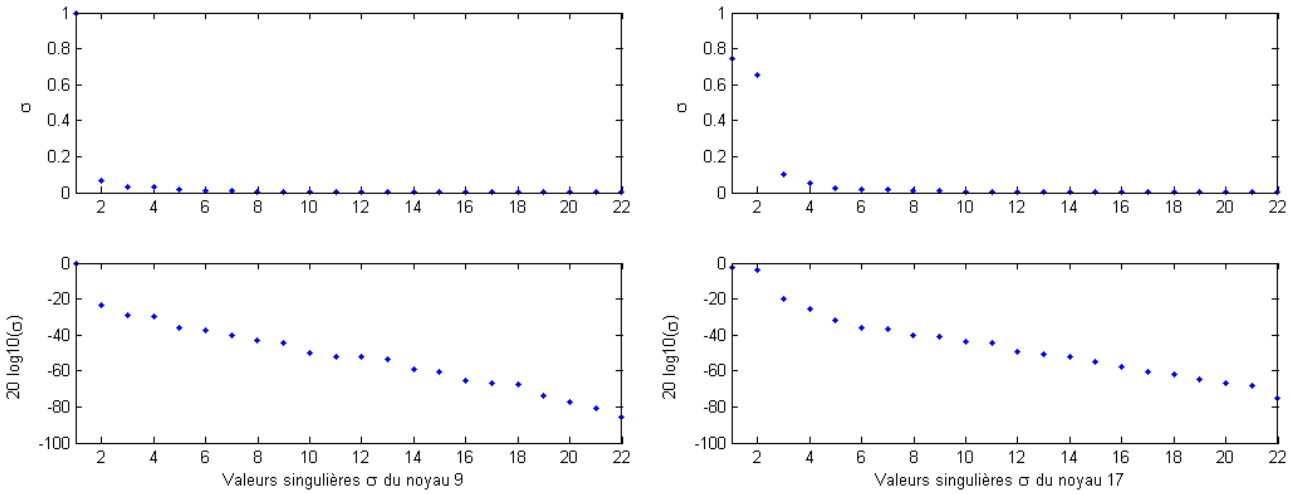


Fig. 2.9 – Valeurs singulières du noyau $l = 9$ (gauche) et du noyau $l = 17$ (droite). Les valeurs sont affichées par ordre décroissant (haut) et en échelle logarithmique (bas).

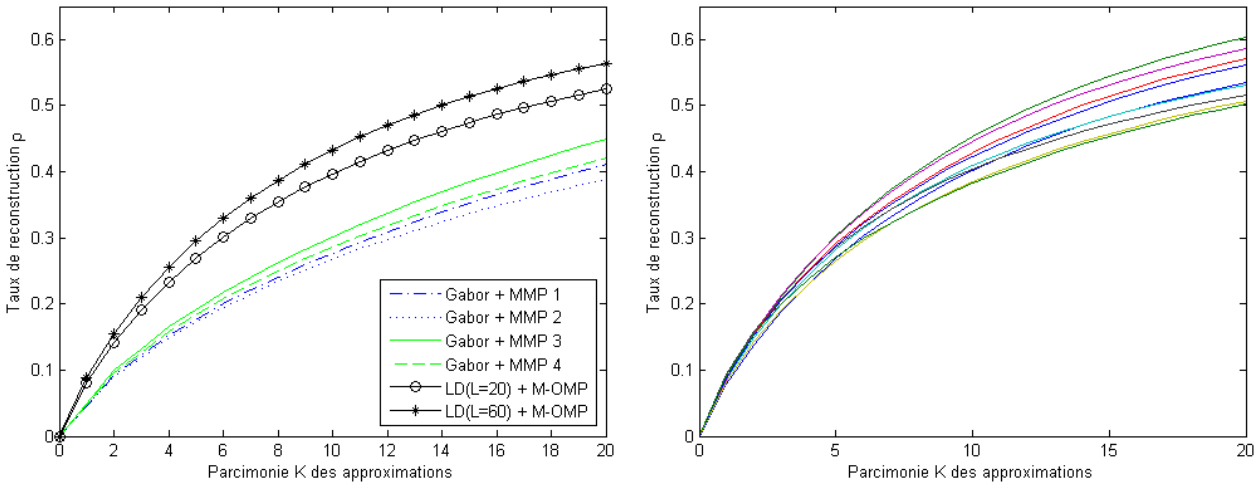
singulières fortes et donc non négligeables. S'il peut être raisonnablement approché par un noyau de rang faible, l'hypothèse de rang 1 n'est pas adaptée ici.
De manière plus générale, les signaux EEG sont constatés comme étant de rangs faibles [RSAG09]. Il est dommageable d'effectuer une hypothèse de rang 1, ce qui supprime une partie de l'information déjà difficile à extraire. Ceci montre les limites du modèle multicanal contrairement au modèle multivarié.

Décompositions et comparaisons

Dans ce paragraphe, la capacité de représentation des différents dictionnaires est évaluée. Dans un premier temps, l'ensemble d'apprentissage du premier sujet est considéré. Les dictionnaires appris (LD pour *Learned Dictionaries*) utilisés avec le M-OMP sont comparés au dictionnaire de Gabor réel utilisé avec le MMP_1 et MMP_2 et au dictionnaire de Gabor complexe utilisé avec le MMP_3 et MMP_4. Les dictionnaires de Gabor ont $M = 30720$ atomes. Deux dictionnaires appris (*learned dictionary* (LD)) sont utilisés : un avec $L = 20$ noyaux (appris précédemment), donnant $M \approx 10000$ atomes ; et un avec $L = 60$, donnant $M \approx 30000$ atomes, ce qui est une taille équivalente aux dictionnaires de Gabor. Pour chaque cas, les approximations K-parcimonieuses sont calculées sur l'ensemble d'apprentissage, et le taux de reconstruction ρ est ensuite calculé. Il est défini comme :

$$\rho = 1 - \frac{1}{P} \sum_{p=1}^{P} \frac{\|\boldsymbol{\epsilon}_p\|}{\|\boldsymbol{y}_p\|} \,. \tag{2.24}$$

Le taux ρ est représenté en fonction de K en Fig. 2.10(a). D'abord, le MMP_1 (courbe bleue trait-pointillée) est meilleur que le MMP_2 (courbe bleue pointillée), et le MMP_3 (courbe verte pleine) que le MMP_4 (courbe verte avec traits), parce que la sélection de l'Eq. (2.3) est plus exigeante que celle de l'Eq. (2.2) comme expliqué en Section 2.5.1. Ensuite, les MMP_3 et MMP_4 sont meilleurs que les MMP_1 et MMP_2 grâce à la flexibilité spatiale des phases des atomes. Enfin, les dictionnaires appris (courbes noires pleines) sont meilleurs que les autres approches, même avec trois fois moins d'atomes. Ces deux représentations (LD avec $L = 20$ et $L = 60$) sont plus compactes étant donné qu'elles sont



(a) (b)

Fig. 2.10 – (a) : taux de reconstruction ρ sur l'ensemble d'apprentissage en fonction de la parcimonie K des approximations des différents dictionnaires et algorithmes. (b) : taux de reconstruction ρ du M-OMP utilisé avec LD ($L = 60$), en fonction de la parcimonie K des approximations sur les 9 ensembles de test.

adaptées aux signaux EEG étudiés. Si les noyaux appris codent plus d'énergie que les atomes de Gabor, le dictionnaire appris prend plus de mémoire (pour le stockage ou la transmission) que le dictionnaire paramétrique de Gabor.

La généralisation de ce résultat est testée dans un deuxième temps, afin de déterminer si cette représentation adaptée, apprise sur l'ensemble d'apprentissage du sujet 1, reste efficace pour d'autres acquisitions et d'autres sujets. Désormais, les ensembles de test des 9 sujets sont considérés. De la même manière, les approximations K-parcimonieuses sont calculées avec le dictionnaire appris LD ($L = 60$) sur les différents ensemble de tests, et les taux de reconstruction ρ sont tracés en fonction de K en Fig. 2.10(b). Les différentes courbes sont similaires et ressemblent à la courbe de LD (courbe noire pleine et étoilée) de la Fig. 2.10(a), ce qui montre la robustesse intra et inter-utilisateurs de la représentation apprise.

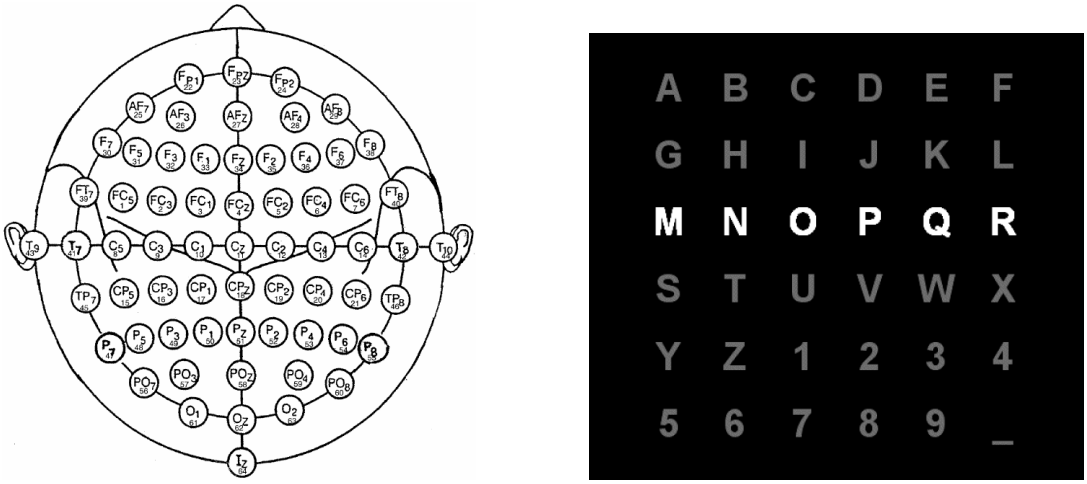
Etant donné qu'ils possèdent de bonnes propriétés de représentation (peu d'atomes codent beaucoup d'énergie), les dictionnaires appris peuvent être utiles pour la simulation de données EEG, prenant en compte leur caractère spatio-temporel.

2.5.3 Experience 2 : apprentissage de potentiels évoqués

L'expérience précédente a montré les performances et la pertinence des représentations adaptées. La question est de savoir si ces noyaux appris peuvent capturer des structures ayant une interprétation physiologique (cf. Section 1.3). Pour y répondre, nous allons étudier le cas de figure des potentiels évoqués P300.

Données P300

Dans cette section, l'expérience est réalisée sur les données 2b du P300 *speller* [BS02], issues de BCI Competition II. Le but du dispositif P300 *speller* est de permettre à un sujet de communiquer grâce à une Interface Cerveau-Machine (ICM) [RSAG09, RJC12] réalisée avec un casque EEG (cf. Fig. 2.11(a)). Une matrice de caractères (lettres et chiffres) représentée en Fig. 2.11(b) s'éclaire périodiquement par colonne ou ligne, envoyant donc des stimuli visuels au sujet. Quand le sujet se concentre sur un des caractères de la matrice et que celui-ci s'éclaire, le stimulus est dit cible. Dans ce cas, un potentiel évoqué appelé P300 est observé dans le cerveau du sujet 300 ms après le stimulus, contrairement à un stimulus non-cible. La difficulté d'un tel dispositif est le très faible RSB des potentiels évoqués P300.



(a) (b)

Fig. 2.11 – Emplacements des électrodes du casque EEG (a), et matrice du dispositif P300 *speller* (b).

Dans ces données, $P = 1261$ stimuli cibles ont lieu, en sachant qu'il y a environ 6 fois plus de stimuli non-cibles. En considérant l'acquisition complète $\boldsymbol{Y} \in \mathbb{R}^{\mathcal{N}\times V}$ de $\mathcal{N}$ échantillons, elle est coupée en P signaux $\{\boldsymbol{y}_p\}_{p=1}^{P}$ de N échantillons (ces signaux sont qualifiés de *epoched* en anglais). $D \in \mathbb{R}^{\mathcal{N}\times N}$ est une matrice de Toeplitz, dont la première colonne est définie telle que $D(t^p, 1) = 1$, où t^p le début du $p^{\text{ième}}$ stimulus cible. Les signaux acquis ont $V = 64$ composantes et sont échantillonnés à 240 Hz. Ils sont filtrés entre 1 et 20 Hz par un filtre de Butterworth d'ordre 3.

Revue des modèles et des méthodes

Il y a plusieurs façons de modéliser le motif P300. En notant $\boldsymbol{\phi}_{\text{P300}} \in \mathbb{R}^{N\times V}$ ce motif, le modèle additif classique s'écrit :

$$\boldsymbol{y} = \boldsymbol{\phi}_{\text{P300}} + \boldsymbol{\epsilon} \,, \tag{2.25}$$

et, avec l'invariance par translation et une amplitude, cela donne un modèle temporel plus flexible [JV99] :

$$\boldsymbol{y}(t) = x\ \boldsymbol{\psi}_{\text{P300}}(t-\tau) + \boldsymbol{\epsilon}(t) \,. \tag{2.26}$$

Ce modèle est adapté pour prendre en compte la variabilité de la latence (délai de réponse au stimulus), aussi appelé *jitter*, des potentiels étudiés. Nous retrouvons l'Eq. (1.4) mais réduite à un seul noyau

($L = 1$) et dans un cas multivarié.

Une manière classique de traiter les P300 consiste à utiliser un motif de forme prédéfinie. Un *a priori* est ainsi injecté à travers ce prototype, généralement utilisé avec un modèle invariant par translation. Par exemple, des motifs monocanaux ont été utilisés pour coïncider avec le P300 ou d'autres potentiels évoqués, comme des fonctions Gaussiennes [LPI97], des sinusoïdes limitées en temps [JV99], des *generic mass potentials* [MBM03], des fonctions Gamma [LPBF09, LKP09] ou encore des fonctions de Gabor [JSM+11]. Des motifs multicanaux ont été introduits dans [GHANZ08] en utilisant des atomes temporels de Gabor et des coefficients spatiaux basés sur des fonctions de Bessel qui tentent de modéliser les dépendances entre canaux. Ces approches s'apparentent au *dictionary design* (cf. Section 1.3.2).

Une autre approche est similaire à l'apprentissage de dictionnaire, en inférant des motifs EEG à partir des données. Différentes méthodes récentes permettent d'apprendre des potentiels évoqués, mais seulement dans le cas monocanal [XSL+09, DSAS11, NFV+12] ou multicanal [KST+06, WG11]. Ici, nous voulons apprendre des motifs multivariés. Considérant le modèle (2.25), Rivet *et al.* [RSAG09] donnent l'estimation des moindres carrés (LS) qui prend en compte les recouvrements qui se produisent suite à des stimuli cibles rapprochés. Elle est donnée par :

$$\begin{aligned} \hat{\boldsymbol{\phi}}_{\text{P300}} &= \arg\min_{\boldsymbol{\phi}} \| \boldsymbol{Y} - D\boldsymbol{\phi} \|_F^2 \\ &= (D^T D)^{-1} D^T \boldsymbol{Y} \,. \end{aligned} \tag{2.27}$$

Cette estimation est optimale s'il n'y a aucune variabilité dans les amplitudes et les latences. De plus, sans recouvrement, cette estimation est équivalente à la grande moyenne, *grand average* (GA) en anglais, effectuée sur les signaux coupés $\{\boldsymbol{y}_p\}_{p=1}^P$:

$$\bar{\boldsymbol{\phi}}_{\text{P300}} = \frac{1}{P} D^T \boldsymbol{Y} = \frac{1}{P} \sum_{p=1}^{P} \boldsymbol{y}_p \,. \tag{2.28}$$

Cependant, ces deux estimations multivariées font de fortes hypothèses sur les latences et les amplitudes, ce qui est un manque de flexibilité. Considérant le modèle (2.26), nous proposons d'utiliser l'apprentissage de dictionnaire, qui peut être vu comme une estimation itérative des moindres carrés :

$$\min_{\boldsymbol{\psi}} \sum_{p=1}^{P} \min_{x_p, \tau_p} \left\| \boldsymbol{y}_p - x_p \, \boldsymbol{\psi}(t - \tau_p) \right\|_F^2 \text{ t.q. } \|\boldsymbol{\psi}\| = 1 \,, \tag{2.29}$$

avec la variable τ restreinte à un intervalle autour de 300 ms. Le noyau estimé est noté $\tilde{\boldsymbol{\psi}}_{\text{P300}}$.

Apprentissage et comparaisons qualitatives

Le M-DLA est appliqué sur l'ensemble d'apprentissage $\{\boldsymbol{y}_p\}_{p=1}^P$, avec $K = 1$. L'estimation par grande moyenne $\bar{\phi}_{\text{P300}}$ est utilisée comme initialisation du noyau, pour fournir un démarrage à chaud, et le M-DLA effectue 20 itérations sur l'ensemble d'apprentissage. La longueur du noyau est de $T = 65$ échantillons, ce qui représente 270 ms, et elle est constante durant l'apprentissage. Le paramètre optimal τ est cherché uniquement sur un intervalle de 9 points centré autour de 300 ms après le stimulus cible[5], ce qui donne une tolérance de latence de $\pm$ 16.7 ms.

5. Cependant, un effet de bord est observé durant l'apprentissage : les décalages temporels τ_p de nombreux signaux sont localisés sur les bords de l'intervalle. Cela signifie que le maximum global de ces corrélations n'a pas été trouvé sur cet

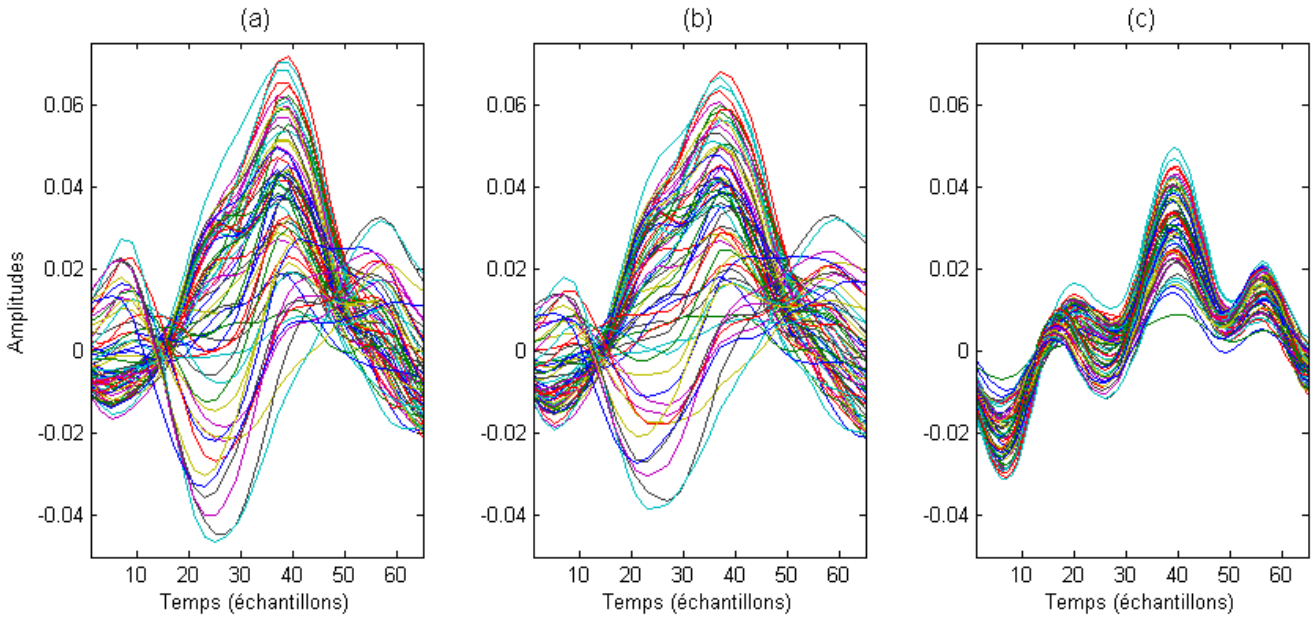


Fig. 2.12 – Motifs temporels multivariés du P300 calculés par grande moyenne (a), par moindres carrées (b) et par apprentissage de dictionnaire multivarié (c). Echantillonnées à 240 Hz, les amplitudes sont données en fonction des échantillons temporels.

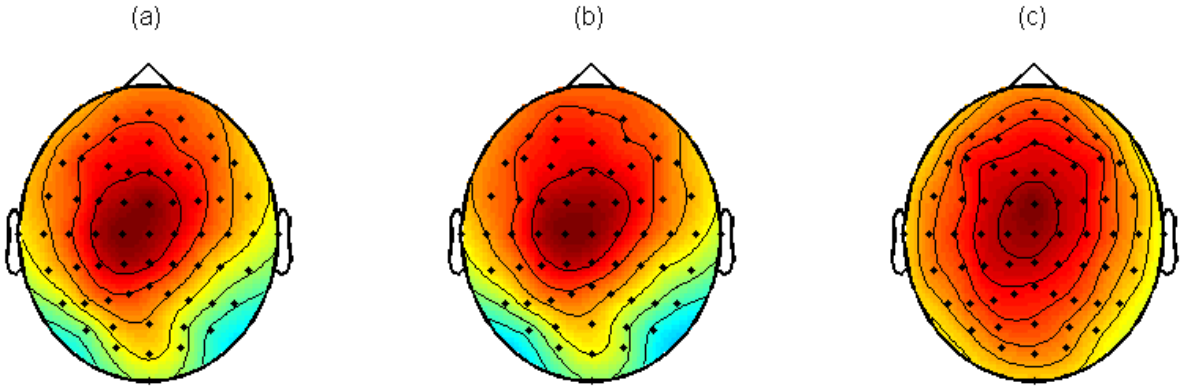


Fig. 2.13 – Motifs spatiaux du P300 calculés par grande moyenne (a), par moindres carrées (b) et par apprentissage de dictionnaire multivarié (c).

Pour être comparées au noyau appris $\tilde{\boldsymbol{\psi}}_{\text{P300}}$, les estimations $\bar{\boldsymbol{\phi}}_{\text{P300}}$ et $\hat{\boldsymbol{\phi}}_{\text{P300}}$ sont d'abord limitées à 65 échantillons puis normées. Elles possèdent $V = 64$ composantes et n'ont pas été filtrées spatialement pour être rehaussées [RSAG09]. Les motifs temporels multivariés sont affichés en Fig. 2.12, avec $\bar{\boldsymbol{\phi}}_{\text{P300}}$ estimé par grande moyenne (a), $\hat{\boldsymbol{\phi}}_{\text{P300}}$ estimé par moindres carrés (b) et $\tilde{\boldsymbol{\psi}}_{\text{P300}}$ par M-DLA (c). L'amplitude est donnée en ordonnée et les échantillons en abscisse. Les motifs (a) et (b) sont similaires, alors que le

intervalle, et que le τ_p est une valeur par défaut. Cela est dû au fort niveau de bruit qui empêche la corrélation de détecter la position du P300. De tels signaux endommageront le noyau $\tilde{\boldsymbol{\psi}}_{\text{P300}}$ s'ils sont utilisés pour la mise à jour du dictionnaire, puisque le paramètre de décalage n'est pas optimal. Pour éviter cela, le noyau n'est pas mis à jour après la décomposition de tels signaux.

noyau (c) est plus fin et les composantes sont en phase.

Les motifs spatiaux associés sont composés des amplitudes du maximum temporel des motifs. Ils sont affichés en Fig. 2.13, où (a) est le motif estimé par GA, (b) par LS et (c) par M-DLA. Nous observons que, de même que pour la comparaison temporelle, les motifs (a) et (b) sont plutôt similaires. Le scalp topographique (c) est plus lisse que les autres et ne possède pas un motif supplémentaire à l'arrière de la tête. Mais, comme la véritable référence du P300 est inconnue, il est difficile d'avoir une comparaison quantitative entre ces différentes estimations.

Comparaison quantitative par analogie

Une solution consiste à valider le cas précédent en procédant par analogie sur un cas simulé. Un motif P300 appris précédemment avec $V = 64$ composantes est choisi pour être le P300 de référence de la simulation. $P = 1000$ signaux sont créés en utilisant ce P300 de référence, avec un paramètre de décalage tiré d'une distribution Gaussienne. Un bruit corrélé spatialement, reproduit avec un filtre FIR appris sur les données EEG [ASS98], est ajouté aux signaux afin d'avoir un RSB de -10 dB. D'abord, l'estimation GA est calculée sur l'ensemble des signaux. Ensuite, cette estimation est utilisée pour l'initialisation du M-DLA, qui est appliqué sur les signaux. Cette expérience est effectuée 50 fois pour différentes valeurs d'écart-type des paramètres de décalage (en nombre d'échantillons). Les motifs estimés à partir de ces deux approches sont comparés quantitativement, en calculant leurs corrélations maximales avec le motif P300 de référence. Etant donné que les motifs sont normés, la valeur absolue des corrélations est comprise entre 0 et 1.

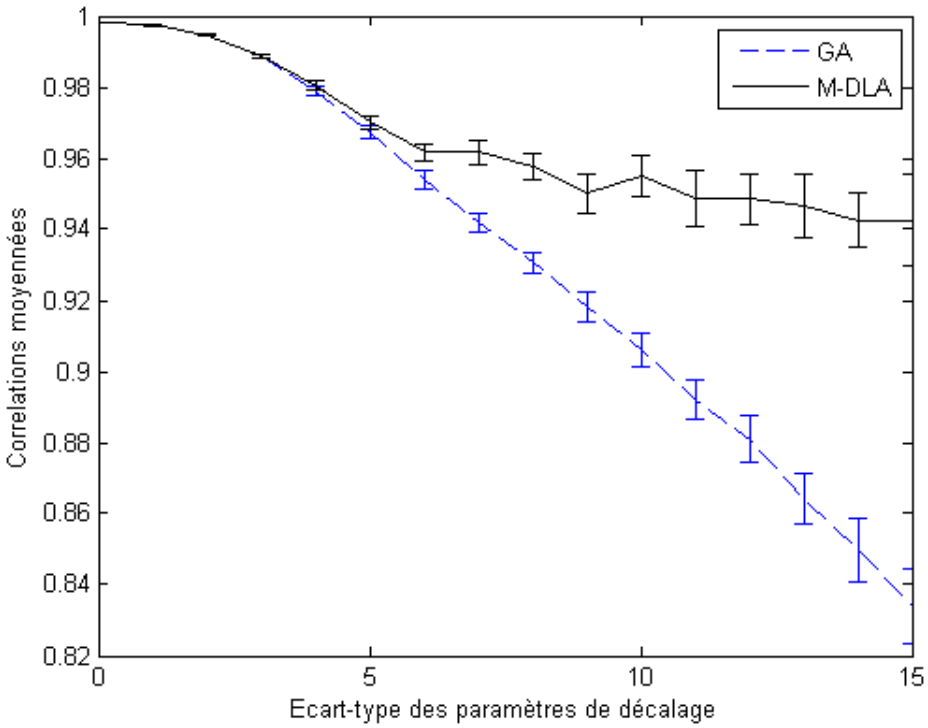


Fig. 2.14 – Corrélations moyennées sur 50 expériences en fonction de l'écart-type des paramètres de décalage (en nombre d'échantillons), pour la grande moyenne (GA) et pour l'apprentissage de dictionnaire multivarié (M-DLA).

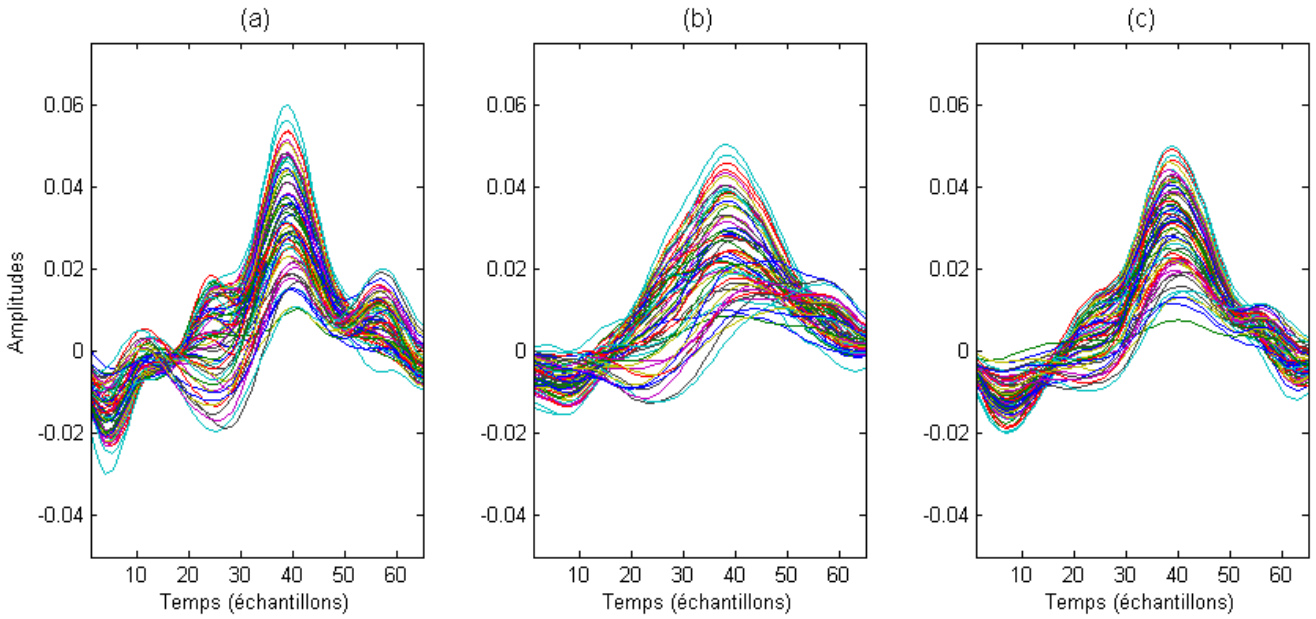


Fig. 2.15 – Motifs temporel du P300 : la référence (a), la grande moyenne (b) et l'apprentissage de dictionnaire multivarié (c).

Les corrélations maximales sont tracées en Fig. 2.14 en fonction de l'écart-type des paramètres de décalage. Si les performances d'identification du GA et du M-DLA sont similaires pour de faibles écart-types, plus les motifs P300 sont décalés, plus le M-DLA surpasse le GA. Cela montre quantitativement que l'apprentissage de dictionnaire invariant par translation est meilleur que l'approche de grande moyenne (équivalente à l'estimation LS dans ce cas puisqu'il n'y a pas de recouvrement entre les signaux).

Les motifs temporels d'une des expériences sont affichés en Fig. 2.15, avec un écart-type $\sigma = 6$. Nous précisons ici que le motif P300 de référence (a) provient de l'apprentissage (cf. Section 2.5.3), mais avec un intervalle de 1. Il est donc seulement estimé avec les signaux donnant leurs corrélations maximales exactement à 300 ms. D'abord, nous observons que les motifs décalés moyennés donnent un motif estimé (b) étalé en comparaison à la référence (a). Nous pouvons montrer que le motif (b) est le résultat d'une convolution temporelle entre la référence (a) et la distribution Gaussienne des paramètres de décalage. Ensuite, le motif M-DLA (c) est plus fin que le (b). Cela confirme les résultats obtenus sur les corrélations. Par analogie, nous pouvons faire l'hypothèse que le P300 de référence est un motif fin, qui est étalé lorsque l'on moyenne des occurrences décalées. Le M-DLA permet d'extraire un motif plus fin grâce à la flexibilité de son invariance par translation.

Comme l'invariance par translation temporelle n'est pas facilement intégrable dans les traitements EEG, l'hypothèse de stationnarité temporelle est souvent faite à travers l'utilisation d'une matrice de covariance [BTL+08] ou par grande moyenne [HCdRM11]. Cette expérience montre que cette hypothèse est grossière et qu'elle provoque une perte d'information temporelle. Bien que le modèle d'invariance par translation et la flexibilité spatiale de notre approche soit une amélioration évidente, le but n'est pas de dire que le noyau P300 appris est meilleur que les autres estimations[6], le but est de présenter une

6. De plus, entre les motifs affichés en Fig. 2.12(c) et 2.15(a), tous les deux estimés par M-DLA, nous ne sommes pas capables de dire lequel est le meilleur.

nouvelle méthode d'estimation de motifs EEG, avec la perspective d'avancer dans la connaissance du P300 et d'améliorer les traitements basés sur l'estimation du P300.

Cette expérience montre que les noyaux appris sont interprétables. Cependant, à cause du bruit élevé, afin d'être interprétés avec un sens physiologique, l'algorithme d'apprentissage doit localiser en temps sur un intervalle les activités d'intérêt comme présenté en Expérience 2, contrairement à l'Expérience 1. Le M-DLA peut être appliqué à d'autres types de potentiels évoqués, comme le N200 ou la négativité de discordance (en anglais *mismatch negativity*).

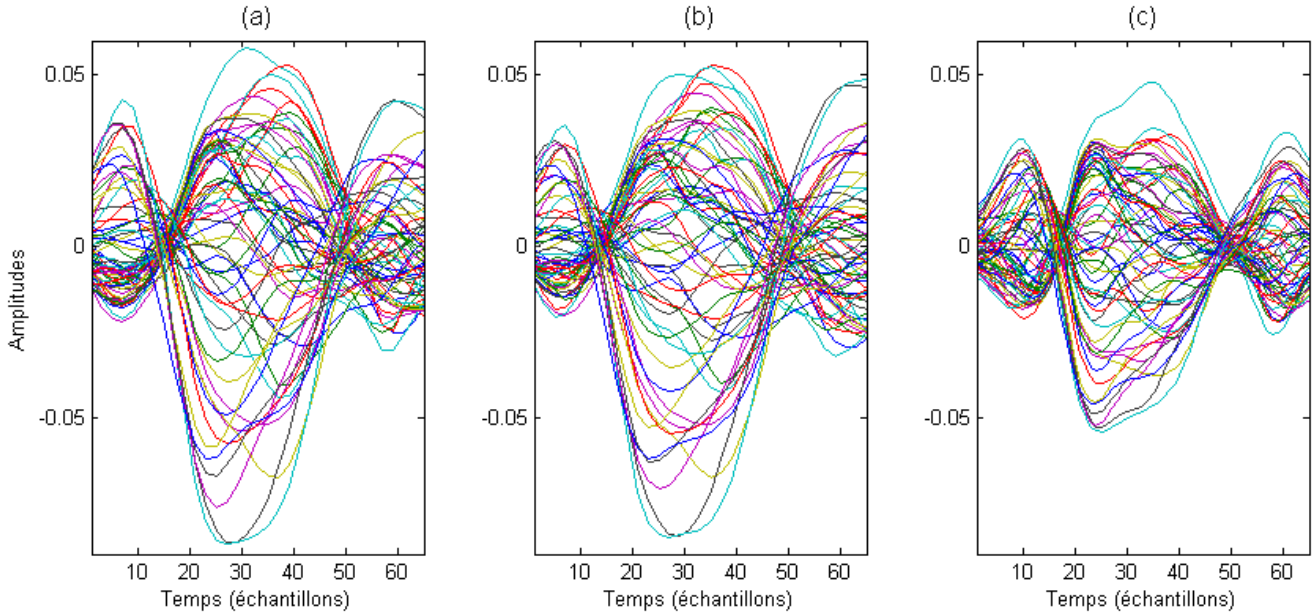


Fig. 2.16 – Motifs temporels multivariés du P300 calculés par grande moyenne (a), par moindres carrées (b) et par apprentissage de dictionnaire multivarié (c), pour la référence d'électrode moyenne.

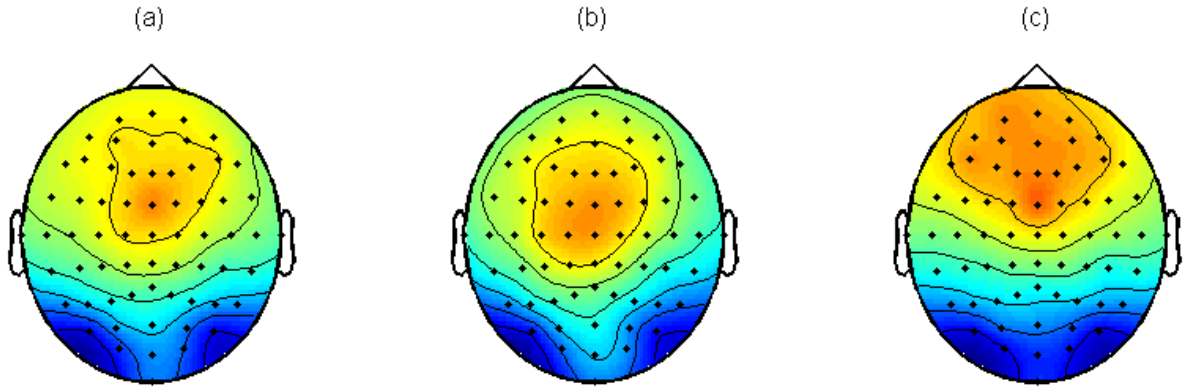


Fig. 2.17 – Motifs spatiaux du P300 calculés par grande moyenne (a), par moindres carrées (b) et par apprentissage de dictionnaire multivarié (c), pour la référence d'électrode moyenne.

Influence de la référence des canaux

Dans les expériences précédentes, le M-DLA semble privilégier des motifs avec des composantes en phase, contrairement aux estimations GA et LS. De plus, comme observé dans [MDMM+05], le choix de la référence des canaux influence les polarités des composantes. Pour mesurer l'influence réelle sur les résultats, l'expérience de la Section 2.5.3 est reproduite avec une référence différente.

Les données utilisées en Section 2.5.3 sont transformées linéairement pour produire une configuration de référence moyenne. Les motifs P300 sont estimés par GA et LS, aussi bien que par M-DLA qui est appliqué avec les mêmes paramètres. Les motifs temporels multivariés du P300 sont affichés en Fig. 2.16 : l'estimation GA (a), l'estimation LS (b) et le M-DLA (c). Nous observons que le M-DLA est capable d'apprendre un motif multivarié avec des phases opposées.

Dans la même manière, les motifs spatiaux du P300 sont affichés en Fig. 2.17 pour GA (a), pour LS (b) et pour le M-DLA (c). Nous observons que les polarités opposées de l'arrière de la tête, et qui n'étaient pas retrouvées en Fig. 2.13(c), sont apprises en Fig. 2.17(c). Ainsi, cette vérification montre que les phases des composantes sont influencées par la référence des canaux, et non par la méthode M-DLA.

Pour conclure, ces deux expériences ont permis d'appliquer nos méthodes sur des données réelles, ainsi que de prouver la pertinence de l'apprentissage de dictionnaire multivarié temporel pour l'analyse EEG. Les particularités de ces signaux, *i.e.* multicomposantes et très bruités, réclament un modèle spatio-temporel spécifique.

Conclusion

Dans ce chapitre, nous avons introduit le modèle multivarié et présenté les méthodes associées : le M-OMP pour réaliser une approximation parcimonieuse multivariée et le M-DLA pour apprendre un dictionnaire multivarié, le tout dans un contexte d'invariance par translation temporelle.

Après avoir comparé nos méthodes dans un cas univarié sur des signaux simulés, nous les avons appliquées à des données EEG. Les résultats montrent la pertinence d'une telle méthode quant à ses qualités de représentation (noyaux informatifs) et à son interprétation (signification physiologique des noyaux pour des activités localisées en temps). Ces travaux ouvrent des perspectives pour les ICM : les méthodes usuelles de classification de signaux EEG [GPCB+10, JCP+11] peuvent être améliorées en prenant en compte cette flexibilité temporelle, et cela ouvre des perspectives de méthodes où l'aspect temporel sera pleinement pris en compte [TWH13]. Comme remarqué dans [TF11a], les dictionnaires appris avec des contraintes de discrimination sont efficaces pour la classification. Les perspectives sont ainsi de modifier la méthode présentée pour donner une approche spatio-temporelle pour les BCI.

Bibliographie

[AE08] M. Aharon et M. Elad : Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM J. Imaging Sciences*, 1:228–247, 2008.

[AEB06] M. Aharon, M. Elad et A.M. Bruckstein : K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54:4311–4322, 2006.

[Aha06] M. Aharon : *Overcomplete Dictionaries for Sparse Representation of Signals.* Thèse de doctorat, Technion - Israel Institute of Technology, 2006.

[ASS98] C.W. Anderson, E.A. Stolz et S. Shamsunder : Multivariate autoregressive models for classification of spontaneous electroencephalographic signals during mental tasks. *IEEE Trans. on Biomedical Engineering*, 45:277–286, 1998.

[BB08] L. Bottou et O. Bousquet : The tradeoffs of large scale learning. *In Advances in Neural Information Processing Systems NIPS*, volume 20, pages 161–168, 2008.

[BDS+05] D. Baron, M.F. Duarte, S. Sarvotham, M.B. Wakin et R.G. Baraniuk : An information-theoretic approach to distributed compressed sensing. *In Proc. Allerton Conf. Communication, Control, and Computing*, 2005.

[BPTC09] C.G. Bénar, T. Papadopoulo, B. Torrésani et M. Clerc : Consensus matching pursuit for multi-trial EEG signals. *J. Neuroscience Methods*, 180:161–170, 2009.

[BS02] B. Blankertz et G. Schalk : 2nd Wadsworth BCI Dataset (P300 Evoked Potentials), 2002.

[BTL+08] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe et K.R. Müller : Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, 41:581–607, 2008.

[CBA13] S. Chevallier, Q. Barthélemy et J. Atif : Metrics for multivariate dictionaries. Preprint arXiv :1302.4242, 2013.

[CGPJ08] M. Congedo, C. Gouy-Pailler et C. Jutten : On the blind source separation of human electroencephalogram by approximate joint diagonalization of second order statistics. *Clinical Neurophysiology*, 119:2677–2686, 2008.

[CH06] J. Chen et X. Huo : Theoretical results on sparse representations of Multiple-Measurement Vectors. *IEEE Trans. on Signal Processing*, 54:4634–4643, 2006.

[CREKD05] S.F. Cotter, B.D. Rao, K. Engan et K. Kreutz-Delgado : Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE Trans. on Signal Processing*, 53:2477–2488, 2005.

[Dau88] I. Daubechies : Time-frequency localization operators : a geometric phase space approach. *IEEE Trans. on Information Theory*, 34:605–612, 1988.

[DB01] P.J. Durka et K.J. Blinowska : A unified time-frequency parametrization of EEGs. *IEEE Engineering in Medicine and Biology Magazine*, 20:47–53, 2001.

[DMMM+05] P.J. Durka, A. Matysiak, E. Martínez-Montes, P.A. Valdés-Sosa et K.J. Blinowska : Multichannel matching pursuit and EEG inverse solutions. *J. Neuroscience Methods*, 148:49–59, 2005.

[DSAS11] C. D'Avanzo, S. Schiff, P. Amodio et G. Sparacino : A Bayesian method to estimate single-trial event-related potentials with application to the study of the P300 variability. *J. Neuroscience Methods*, 198:114–124, 2011.

[Dur07] P.J. Durka : *Matching Pursuit and Unification in EEG Analysis.* Artech House, 2007.

[ER10] Y.C. Eldar et H. Rauhut : Average case analysis of multichannel sparse recovery using convex relaxation. *IEEE Trans. on Information Theory*, 56:505–519, 2010.

[GHANZ07] M. Gratkowski, J. Haueisen, L. Arendt-Nielsen et F. Zanow : Topographic matching pursuit of spatio-temporal bioelectromagnetic data. *Przeglad Elektrotechniczny*, 83:138–141, 2007.

[GHANZ08] M. Gratkowski, J. Haueisen, A. Arendt-Nielsen, L. Chen et F. Zanow : Decomposition of biomedical signals in spatial and time-frequency modes. *Methods of Information in Medicine*, 47:26–37, 2008.

[GN05] R. Gribonval et M. Nielsen : Beyond sparsity : Recovering structured representations by ℓ_1-minimization and greedy algorithms. -Application to the analysis of sparse underdetermined ICA-. Rapport technique PI-1684, IRISA, 2005.

[GPCB+10] C. Gouy-Pailler, M. Congedo, C. Brunner, C. Jutten et G. Pfurtscheller : Nonstationary brain source separation for multiclass motor imagery. *IEEE Trans. on Biomedical Engineering*, 57:469–478, 2010.

[Gri02] R. Gribonval : Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '02*, pages 3057–3060, 2002.

[Gri03] R. Gribonval : Piecewise linear source separation. *In Proc. SPIE 5207*, pages 297–310, 2003.

[GRSV07] R. Gribonval, H. Rauhut, K. Schnass et P. Vandergheynst : Atoms of all channels, unite ! Average case analysis of multi-channel sparse recovery using greedy algorithms. Rapport technique PI-1848, IRISA, 2007.

[HCdRM11] B. Hamner, R. Chavarriaga et J. del R. Millán : Learning dictionaries of spatial and temporal EEG primitives for brain-computer interfaces. *In Workshop on structured sparsity : learning and inference ; ICML 2011*, 2011.

[JCP+11] N. Jrad, M. Congedo, R. Phlypo, S. Rousseau, R. Flamary, F. Yger et A. Rakotomamonjy : sw-SVM : sensor weighting support vector machines for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 8:1–11, 2011.

[JSM+11] M. Jörn, C. Sielużycki, M.A. Matysiak, J. Żygierewicz, H. Scheich, P.J. Durka et R. König : Single-trial reconstruction of auditory evoked magnetic fields by means of template matching pursuit. *J. Neuroscience Methods*, 199:119–128, 2011.

[JV99] P. Jaśkowski et R. Verleger : Amplitudes and latencies of single-trial ERP's estimated by a maximum-likelihood method. *IEEE Trans. on Biomedical Engineering*, 46:987–993, 1999.

[JVLG05] P. Jost, P. Vandergheynst, S. Lesage et R. Gribonval : Learning redundant dictionaries with translation invariance property : the MoTIF algorithm. *In Workshop Structure et parcimonie pour la représentation adaptative de signaux SPARS '05*, 2005.

[KMLVS01] T. Koenig, F. Marti-Lopez et P. Valdés-Sosa : Topographic time-frequency decomposition of the EEG. *NeuroImage*, 14:383–390, 2001.

[KST+06] K.H. Knuth, A.S. Shah, W.A. Truccolo, M. Ding, S.L. Bressler et C.E. Schroeder : Differentially variable component analysis : Identifying multiple evoked components using trial-to-trial variability. *Journal of Neurophysiology*, 95:3257–3276, 2006.

[LKG06] S. Lesage, S. Krstulović et R. Gribonval : Under-determined source separation : comparison of two approaches based on sparse decompositions. *In Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation*, pages 633–640, 2006.

[LKP09] R. Li, A. Keil et J.C. Principe : Single-trial P300 estimation with a spatiotemporal filtering method. *J. Neuroscience Methods*, 177:488–496, 2009.

[LPBF09] R. Li, J.C. Principe, M. Bradley et V. Ferrari : A spatiotemporal filtering methodology for single-trial ERP component estimation. *IEEE Trans. on Biomedical Engineering*, 56:83–92, 2009.

[LPI97] D.H. Lange, H. Pratt et G.F. Inbar : Modeling and estimation of single evoked brain potential components. *IEEE Trans. on Biomedical Engineering*, 44:791–799, 1997.

[LT03a] D. Leviathan et V.N. Temlyakov : Simultaneous approximation by greedy algorithms. Rapport technique, Univ. of South Carolina at Columbia, 2003.

[LT03b] A. Lutoborski et V.N. Temlyakov : Vector greedy algorithms. *J. Complex.*, 19:458–473, 2003.

[LT05] D. Leviathan et V.N. Temlyakov : Simultaneous greedy approximation in Banach spaces. *J. Complex.*, 21:275–293, 2005.

[Mal09] S. Mallat : *A Wavelet Tour of signal processing.* 3rd edition, New-York : Academic, 2009.

[MBM03] D. Melkonian, T.D. Blumenthal et R. Meares : High-resolution fragmentary decomposition - a model-based method of non-stationary electrophysiological signal analysis. *J. Neuroscience Methods*, 131:149–159, 2003.

[MBPS10] J. Mairal, F. Bach, J. Ponce et G. Sapiro : Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010.

[MBSF09] Y. Moudden, J. Bobin, J.-L. Starck et J. Fadili : Dictionary learning with spatio-spectral sparsity constraints. *In Signal Processing with Adaptive Sparse Structured Representations SPARS '09*, 2009.

[MDMM+05] A. Matysiak, P.J. Durka, E. Martínez-Montes, M. Barwiński, P. Zwoliński, M. Roszkowski et K.J. Blinowska : Time-frequency-space localization of epileptic EEG oscillations. *Acta Neurobiol. Exp.*, 65:435–442, 2005.

[MES08] J. Mairal, M. Elad et G. Sapiro : Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17:53–69, 2008.

[MGB+09] B. Mailhé, R. Gribonval, F. Bimbot, M. Lemay, P. Vandergheynst et J.M. Vesin : Dictionary learning for the sparse modelling of atrial fibrillation in ECG signals. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '09*, pages 465–468, 2009.

[MHA+08] M. Mørup, L.K. Hansen, S.M. Arnfred, L.-H. Lim et K.H. Madsen : Shift-invariant multilinear decomposition of neuroimaging data. *NeuroImage*, 42:1439–1450, 2008.

[MJV+07] G. Monaci, P. Jost, P. Vandergheynst, B. Mailhé, S. Lesage et R. Gribonval : Learning multimodal dictionaries. *IEEE Trans. on Image Processing*, 16:2272–2283, 2007.

[MNT04] K. Madsen, H.B. Nielsen et O. Tingleff : Methods for non-linear least squares problems, 2nd edition. Rapport technique, Technical University of Denmark, 2004.

[MVS09] G. Monaci, P. Vandergheynst et F.T. Sommer : Learning bimodal structure in audio-visual data. *IEEE Trans. on Neural Networks*, 20:1898–1910, 2009.

[NFV+12] A. Nonclercq, M. Foulon, D. Verheulpen, C. De Cock, M. Buzatu, P. Mathys et P. Van Bogaert : Cluster-based spike detection algorithm adapts to interpatient and intrapatient variation in spike morphology. *J. Neuroscience Methods*, 210:259–265, 2012.

[NLdS04] E. Niedermeyer et F.H. Lopes da Silva : *Electroencephalography : Basic principles, clinical applications and related fields.* 5th edition, 2004.

[NSL+12a] Q. Noirhomme, A. Soddu, R. Lehembre, M.-A. Bruno, D. Lesenfants, S. Laureys et F. Gomez : A dictionary learning approach to study disorders of consciousness : a preliminary EEG study. *In Belgian Brain Council*, 2012.

[NSL+12b] Q. Noirhomme, A. Soddu, R. Lehembre, M.-A. Bruno, D. Lesenfants, S. Laureys et F. Gómez : Dictionary learning in patients with disorders of consciousness : a preliminary EEG study. *In Human Brain Mapping Conference*, 2012.

[PMML95] R.D. Pascual-Marqui, C.M. Michel et D. Lehmann : Segmentation of brain electrical activity into microstates : Model estimation and validation. *IEEE Trans. on Biomedical Engineering*, 42:658–665, 1995.

[PP08] K.B. Petersen et M.S. Pedersen : The matrix cookbook. Rapport technique, Technical University of Denmark, 2008.

[PRK93] Y.C. Pati, R. Rezaiifar et P.S. Krishnaprasad : Orthogonal Matching Pursuit : recursive function approximation with applications to wavelet decomposition. *In Conf. Record of the Asilomar Conf. on Signals, Systems and Comput.*, pages 40–44, 1993.

[Rak11] A. Rakotomamonjy : Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Signal Process.*, 91:1505–1526, 2011.

[RJC12] S. Rousseau, C. Jutten et M. Congedo : Closed-looping a P300 BCI using the ErrP. *In Proc. Int. Conf. Neural Computation Theory and Applications NCTA 2012*, pages 732–737, 2012.

[RSAG09] B. Rivet, A. Souloumiac, V. Attina et G. Gibert : xDAWN algorithm to enhance evoked potentials : Application to brain-computer interface. *IEEE Trans. on Biomedical Engineering*, 56: 2035–2043, 2009.

[SC07] S. Sanei et J. Chambers : *EEG signal processing.* John Wiley & Sons, 2007.

[SE10] K. Skretting et K. Engan : Recursive least squares dictionary learning algorithm. *IEEE Trans. on Signal Processing*, 58:2121–2130, 2010.

[SHA06] K. Skretting, J.H. Husøy et S.O. Aase : General design algorithm for sparse frame expansions. *Signal Process.*, 86:117–126, 2006.

[SKM+09a] C. Sielużycki, R. König, A. Matysiak, R. Kuś, D. Ircha et P.J. Durka : Single-trial evoked brain responses modeled by multivariate matching pursuit. *IEEE Trans. on Biomedical Engineering*, 56:74–82, 2009.

[SKM+09b] C. Sielużycki, R. Kuś, A. Matysiak, P.J. Durka et R. König : Multivariate matching pursuit in the analysis of single-trial latency of the auditory M100 acquired with MEG. *Int. Journal of Bioelectromagnetism*, 11:155–160, 2009.

[SP11] S. Scholler et H. Purwins : Sparse approximations for drum sound classification. *IEEE Journal of Selected Topics in Signal Processing*, 5:933–940, 2011.

[TF11a] I. Tošić et P. Frossard : Dictionary learning. *IEEE Signal Processing Magazine*, 28:27–38, 2011.

[TF11b] I. Tošić et P. Frossard : Dictionary learning for stereo image representation. *IEEE Trans. on Image Processing*, 20:921–934, 2011.

[TGS06] J.A. Tropp, A.C. Gilbert et M.J. Strauss : Algorithms for simultaneous sparse approximation ; Part I : Greedy pursuit. *Signal Process. - Sparse approximations in signal and image processing*, 86:572–588, 2006.

[TMA+12] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K.J. Miller, G. Müller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert et B. Blankertz : Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6:1–31, 2012.

[Tro06] J.A. Tropp : Algorithms for simultaneous sparse approximation ; Part II : Convex relaxation. *Signal Process. - Sparse approximations in signal and image processing*, 86:589–602, 2006.

[TWH13] D.E. Thompson, S. Warschausky et J.E. Huggins : Classifier-based latency estimation : a novel way to estimate and predict BCI accuracy. *Journal of Neural Engineering*, 10:1–7, 2013.

[WG11] W. Wu et S. Gao : Learning event-related potentials (ERPs) from multichannel EEG recordings : a spatio-temporal modeling framework with a fast estimation algorithm. *In Int. Conf. IEEE EMBS*, pages 6959–6962, 2011.

[XSL+09] L. Xu, P. Stoica, J. Li, S.L. Bressler, X. Shao et M. Ding : ASEO : A method for the simultaneous estimation of single-trial event-related potentials and ongoing brain activities. *IEEE Trans. on Biomedical Engineering*, 56:111–121, 2009.

Chapitre 3

Représentations parcimonieuses quaternioniques

Introduction

Nous allons maintenant étendre les méthodes d'approximations parcimonieuses aux signaux quaternioniques [Zha97]. Ce type de signaux permet de traiter des données trivariées et quadrivariées, telles que des images couleur [MSE03, ES07], des ondes polarisées [MLBM06], des signaux EEG [JTJ+11], etc. De plus, les quaternions sont adaptés pour la rotation 3D [Han06]. Cette extension est réalisée avec l'idée de pouvoir effectuer des rotations 3D avec des noyaux trivariés.

Le modèle linéaire (1.1) entre le dictionnaire et les coefficients doit être reconsidéré avec attention car l'espace des quaternions est non commutatif : nous étudierons donc deux modèles linéaires en fonction de l'ordre de la multiplication. Nous donnerons donc deux extensions de l'algorithme OMP, et nous les appliquerons sur des données de simulation.

3.1 Modèles linéaires quaternioniques

Après une courte introduction de l'algèbre des quaternions et des éléments nécessaires à nos développements, nous exposerons les deux modèles linéaires en fonction de l'ordre de la multiplication : à droite ou à gauche.

3.1.1 Algèbre des quaternions

L'ensemble des quaternions noté $\mathbb{H}$, est une extension de l'ensemble des complexes $\mathbb{C}$ en utilisant trois parties imaginaires [Zha97, Han06]. Un quaternion $\mathfrak{q} \in \mathbb{H}$ est défini comme :

$$\mathfrak{q} = \mathfrak{q}_a + \mathfrak{q}_b \mathsf{i} + \mathfrak{q}_c \mathsf{j} + \mathfrak{q}_d \mathsf{k}\,, \tag{3.1}$$

avec $\mathfrak{q}_a$, $\mathfrak{q}_b$, $\mathfrak{q}_c$, $\mathfrak{q}_d \in \mathbb{R}$ et avec les unités imaginaires définies comme :

$$\mathsf{ij} = \mathsf{k},\ \mathsf{jk} = \mathsf{i},\ \mathsf{ki} = \mathsf{j}, \tag{3.2}$$

$$\mathsf{ji} = -\mathsf{k},\ \mathsf{kj} = -\mathsf{i},\ \mathsf{ik} = -\mathsf{j}, \tag{3.3}$$

$$\text{et}\ \ \mathsf{i}^2 = \mathsf{j}^2 = \mathsf{k}^2 = \mathsf{ijk} = -1\,. \tag{3.4}$$

La partie scalaire est $\mathcal{S}(\mathfrak{q}) = \mathfrak{q}_a$, et la partie vectorielle est $\mathcal{V}(\mathfrak{q}) = \mathfrak{q}_b\mathsf{i} + \mathfrak{q}_c\mathsf{j} + \mathfrak{q}_d\mathsf{k}$. Par défaut un quaternion est dit plein et, si sa partie scalaire est nulle, il est dit pur. Le conjugué $\mathfrak{q}^*$ est défini comme : $\mathfrak{q}^* = \mathcal{S}(\mathfrak{q}) - \mathcal{V}(\mathfrak{q})$ et nous avons $(\mathfrak{q}_1\mathfrak{q}_2)^* = \mathfrak{q}_2^*\mathfrak{q}_1^*$. Le module est défini comme $|\mathfrak{q}| = \sqrt{\mathfrak{q}\mathfrak{q}^*}$ et l'inverse comme $\mathfrak{q}^{-1} = \mathfrak{q}^*/|\mathfrak{q}|^2$.

L'espace des quaternions est caractérisé par sa non-commutativité : généralement $\mathfrak{q}_1\mathfrak{q}_2 \neq \mathfrak{q}_2\mathfrak{q}_1$. Deux modèles linéaires différents sont donc possibles : celui avec les coefficients multipliés à droite, et celui avec les coefficients multipliés à gauche. Par la suite, nous détaillerons chaque modèle linéaire, doté de son produit scalaire spécifique.

Un signal quaternionique $y \in \mathbb{H}^N$ permet donc de traiter un signal réel quadrivarié $\boldsymbol{y} \in \mathbb{R}^{N\times 4}$, ou trivarié $\boldsymbol{y} \in \mathbb{R}^{N\times 3}$ en ne considérant que la partie vectorielle du signal quaternionique. De plus, nous définissons l'opérateur transconjugué $(.)^H$ comme : $y^H = \mathcal{S}(y)^T - \mathcal{V}(y)^T$.

3.1.2 Modèle linéaire avec multiplication à droite

Le modèle linéaire étudié est l'extension quaternionique naturelle du modèle linéaire réel ou complexe (1.1). Nous étudions le signal quaternionique $y \in \mathbb{H}^N$ composé de N échantillons, et le dictionnaire $\Phi \in \mathbb{H}^{N\times M}$ composé de M atomes $\{\phi_m\}_{m=1}^M$. La décomposition linéaire du signal y est effectuée sur le dictionnaire Φ telle que :

$$y = \Phi x + \epsilon \tag{3.5}$$

$$= \sum_{m=1}^{M} \phi_m \, x_m + \epsilon \,, \tag{3.6}$$

avec $x \in \mathbb{H}^M$ le vecteur de coefficients et $\epsilon \in \mathbb{H}^N$ l'erreur résiduelle. Ce modèle est utilisé dans plusieurs applications telles que la prévision de vent [TM10], le débruitage d'images couleur [TM10] et la séparation de sources de mélange de signaux EEG [JTJ+11].

Considérant les vecteurs quaternioniques $\mathfrak{q}_1$, $\mathfrak{q}_2 \in \mathbb{H}^N$, nous définissons le *produit scalaire à droite* (cf. Annexe 7.4) comme : $\langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_{\mathsf{r}} = \mathfrak{q}_2^H\mathfrak{q}_1 \in \mathbb{H}$, avec la norme ℓ_2 associée notée $\|.\|_2$. Une définition alternative peut être choisie : $\langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle = \mathfrak{q}_1^H\mathfrak{q}_2$, qui est seulement la conjuguée de la précédente[1]. Le problème d'approximation parcimonieuse pour ce modèle est donc la simple extension quaternionique de l'Eq. (1.6) :

$$\min_x \ \| y - \Phi x \|_2^2 \ \text{ t.q. } \ \|x\|_0 \leq K \,. \tag{3.7}$$

3.1.3 Modèle linéaire avec multiplication à gauche

Pour ce modèle de multiplication à gauche, les différentes variables doivent être transposées : le signal étudié est disposé horizontalement. Nous étudions donc le signal quaternionique $y \in \mathbb{H}^{1\times N}$ composé de N échantillons, et le dictionnaire $\Phi \in \mathbb{H}^{M\times N}$ composé de M atomes $\{\phi_m\}_{m=1}^M$. La décomposition linéaire

1. Notons aussi que le produit scalaire de $\mathbb{R}^{N\times 4}$: $\langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle = \mathcal{S}(\mathfrak{q}_1^H\mathfrak{q}_2) \in \mathbb{R}$ n'est pas considéré ici, bien que souvent utilisé pour traiter les quaternions.

du signal y est effectuée sur le dictionnaire Φ telle que :

$$y = x\,\Phi + \epsilon \tag{3.8}$$

$$= \sum_{m=1}^{M} x_m\,\phi_m + \epsilon\,, \tag{3.9}$$

avec $x \in \mathbb{H}^{1\times M}$ le vecteur de coefficients et $\epsilon \in \mathbb{H}^{1\times N}$ l'erreur résiduelle. Ce modèle est utilisé par exemple pour le débruitage d'images couleur [AC12] sur un dictionnaire utilisant des polynômes quaternioniques.

Pour ce modèle, nous définissons le *produit scalaire à gauche* (cf. Annexe 7.4) comme : $\langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_{\mathsf{l}} = \mathfrak{q}_1 \mathfrak{q}_2^H \in \mathbb{H}$, et la norme ℓ_2 associée est notée $\|.\|_2$. Le problème d'approximation parcimonieuse pour ce modèle s'écrit donc :

$$\min_x \ \| y - x\,\Phi \|_2^2 \ \text{ t.q. } \ \|x\|_0 \leq K\,. \tag{3.10}$$

Remarquons le modèle multiplicatif à gauche (3.8) peut être réécrit en un modèle à droite, après transconjugaison des éléments :

$$y^H = \Phi^H\,x^H + \epsilon^H\,. \tag{3.11}$$

A cause de la non commutativité des quaternions, il y a deux modèles linéaires : nous exposerons donc les deux algorithmes correspondants, même si la transformation de l'un à l'autre est simple.

3.2 OMP quaternioniques

Dans cette section, nous détaillons les deux algorithmes résolvant les problèmes d'approximations parcimonieuses (3.7) et (3.10). Etant basés sur l'OMP, ils sont appelés OMP quaternioniques, soit *Quaternionic* OMP (Q-OMP) en anglais. Dans les deux cas, le dictionnaire utilisé est normé. A cause de la non commutativité, l'ordre des variables est important dans la description des algorithmes. Par souci de simplicité, les deux algorithmes sont présentés en parallèle : le QOMP à gauche résout le problème (3.10) et est présenté dans la colonne de gauche, et le QOMP à droite résout le problème (3.7) et est présenté dans la colonne de droite. Les principales modifications par rapport à l'OMP (Algorithme 1) concernent la sélection (étape 4) et la projection orthogonale (étape 8).

3.2.1 Description du Q-OMP à gauche (*resp.* à droite)

Le Q-OMP à gauche (*resp.* à droite), abrégé en Q-OMPl avec l pour *left* (*resp.* Q-OMPr avec r pour *right*), résout le problème (3.10) (*resp.* problème (3.7)). Il est décrit dans l'Algorithme 8 (*resp.* Algorithme 9).

A l'étape 4, comme dans le cas réel, le produit scalaire reste l'expression à maximiser pour sélectionner l'atome optimal (cf. Annexe 7.6 (*resp.* Annexe 7.5)) mais il faut, pour cela, utiliser le produit scalaire quaternionique spécifique défini en Section 3.1.3 : $\left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle_{\mathsf{l}} = \epsilon^{k-1}\phi_m^H$ (*resp.* Section 3.1.2 : $\left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle_{\mathsf{r}} = \phi_m^H\epsilon^{k-1}$). Nous rappelons qu'à l'itération k, la corrélation optimale est notée $C^k_{m^k}$. De plus, l'atome optimal ϕ_{m^k} qui est intégré au dictionnaire actif D^k est noté d_k.

A l'étape 8, les coefficients x^k sont calculés par projection orthogonale du signal y sur le dictionnaire actif $D^k \in \mathbb{H}^{k\times N}$ (*resp.* $D^k \in \mathbb{H}^{N\times k}$) :

$$\begin{aligned} x^k &= \arg\min_x \left\| y - x\, D^k \right\|_2^2 \\ &= y\,(D^k)^H \left(D^k\,(D^k)^H\right)^{-1}. \end{aligned} \tag{3.12}$$

$$\begin{aligned} x^k &= \arg\min_x \left\| y - D^k x \right\|_2^2 \\ &= \left((D^k)^H\, D^k\right)^{-1} (D^k)^H y\,. \end{aligned} \tag{3.13}$$

Pour calculer cette projection, la procédure récursive d'inversion matricielle par bloc [PRK93] est étendue aux quaternions et avec la multiplication à gauche (*resp.* à droite). Par la suite, la matrice A_k est définie comme la matrice de Gram du dictionnaire actif D^{k-1} :

$$\begin{aligned} A_k &= D^{k-1}\,(D^{k-1})^H \\ &= \begin{bmatrix} \langle d_1, d_1\rangle_{\mathsf{l}} & \langle d_1, d_2\rangle_{\mathsf{l}} & \dots & \langle d_1, d_{k-1}\rangle_{\mathsf{l}} \\ \langle d_2, d_1\rangle_{\mathsf{l}} & \langle d_2, d_2\rangle_{\mathsf{l}} & \dots & \langle d_2, d_{k-1}\rangle_{\mathsf{l}} \\ \vdots & \vdots & \ddots & \vdots \\ \langle d_{k-1}, d_1\rangle_{\mathsf{l}} & \langle d_{k-1}, d_2\rangle_{\mathsf{l}} & \dots & \langle d_{k-1}, d_{k-1}\rangle_{\mathsf{l}} \end{bmatrix}. \end{aligned} \tag{3.14}$$

$$\begin{aligned} A_k &= (D^{k-1})^H\, D^{k-1} \\ &= \begin{bmatrix} \langle d_1, d_1\rangle_{\mathsf{r}} & \langle d_2, d_1\rangle_{\mathsf{r}} & \dots & \langle d_{k-1}, d_1\rangle_{\mathsf{r}} \\ \langle d_1, d_2\rangle_{\mathsf{r}} & \langle d_2, d_2\rangle_{\mathsf{r}} & \dots & \langle d_{k-1}, d_2\rangle_{\mathsf{r}} \\ \vdots & \vdots & \ddots & \vdots \\ \langle d_1, d_{k-1}\rangle_{\mathsf{r}} & \langle d_2, d_{k-1}\rangle_{\mathsf{r}} & \dots & \langle d_{k-1}, d_{k-1}\rangle_{\mathsf{r}} \end{bmatrix}. \end{aligned} \tag{3.15}$$

La diagonale est égale à 1 puisque le dictionnaire est normé. A l'itération k, la procédure récursive pour la projection orthogonale est calculée en sept étapes :

$$\begin{aligned} 1: \quad v_k &= D^{k-1}\, d_k^H \\ &= [\,\langle d_1, d_k\rangle_{\mathsf{l}}\,;\, \langle d_2, d_k\rangle_{\mathsf{l}} \dots \langle d_{k-1}, d_k\rangle_{\mathsf{l}}\,], \\ 2: \quad b_k &= v_k^H A_k^{-1}, \\ 3: \quad \beta &= 1/(1 - b_k v_k), \\ 4: \quad \alpha_k &= C^k_{m^k} \cdot \beta^*. \end{aligned} \tag{3.16}$$

$$\begin{aligned} 1: \quad v_k &= (D^{k-1})^H\, d_k \\ &= [\,\langle d_k, d_1\rangle_{\mathsf{r}}\,;\, \langle d_k, d_2\rangle_{\mathsf{r}} \dots \langle d_k, d_{k-1}\rangle_{\mathsf{r}}\,], \\ 2: \quad b_k &= A_k^{-1} v_k, \\ 3: \quad \beta &= 1/(\|d_k\|_2^2 - v_k^H b_k) = 1/(1 - v_k^H b_k), \\ 4: \quad \alpha_k &= C^k_{m^k} \cdot \beta. \end{aligned} \tag{3.17}$$

Pour fournir la projection orthogonale, les coefficients x_{m^κ} ($\kappa = 1\,..\,k-1$) du vecteur x^k sont corrigés à chaque itération. Nous ajoutons un exposant aux coefficients x_{m^κ} pour indicer l'itération, et la mise à jour est :

$$\begin{aligned} 5: \quad x^k_{m^\kappa} &= x^{k-1}_{m^\kappa} - \alpha_k b_k, \text{ pour } \kappa = 1\,..\,k-1, \\ 6: \quad x^k_{m^k} &= \alpha_k. \end{aligned} \tag{3.18}$$

$$\begin{aligned} 5: \quad x^k_{m^\kappa} &= x^{k-1}_{m^\kappa} - b_k \alpha_k, \text{ pour } \kappa = 1\,..\,k-1, \\ 6: \quad x^k_{m^k} &= \alpha_k. \end{aligned} \tag{3.19}$$

La matrice de Gram est mise à jour telle que :

$$A_{k+1} = \left[\begin{array}{c|c} A_k & v_k \\ \hline v_k^H & 1 \end{array}\right] \tag{3.20}$$

et en utilisant la formule d'inversion matricielle par bloc, nous obtenons son inverse à droite (*resp.* à gauche) :

$$7: \quad A_{k+1}^{-1} = \left[\begin{array}{c|c} A_k^{-1} + \beta b_k^H b_k & -\beta b_k^H \\ \hline -\beta b_k & \beta \end{array}\right]. \tag{3.21}$$

$$7: \quad A_{k+1}^{-1} = \left[\begin{array}{c|c} A_k^{-1} + \beta b_k b_k^H & -\beta b_k \\ \hline -\beta b_k^H & \beta \end{array}\right]. \tag{3.22}$$

Pour la première itération, la procédure est réduite à : $x^1_{m^1} = C^1_{m^1}$ et $A_1 = 1$.

Avec les modifications décrites, le Q-OMPl (*resp.* Q-OMPr) fournit l'approximation K-parcimonieuse du signal y :

$$\hat{y}^K = \sum_{k=1}^{K} x_{m^k}\, \phi_{m^k} \,. \qquad (3.23) \qquad\qquad \hat{y}^K = \sum_{k=1}^{K} \phi_{m^k}\, x_{m^k} \,. \qquad (3.24)$$

Algorithm 8 : $x = \text{Q-OMPl}\ (y, \Phi)$

1: **initialisation** : $k = 1,\ \epsilon^0 = y,$ dictionnaire $D^0 = \varnothing$
2: **repeat**
3: **for** $m \leftarrow 1, M$ **do**
4: Produits scalaires : $C^k_m \leftarrow \left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle_{\mathsf{l}} = \epsilon^{k-1}\, \phi^H_m$
5: **end for**
6: Sélection : $m^k \leftarrow \arg\max_m \,\left|\, C^k_m \,\right|$
7: Dictionnaire actif : $D^k \leftarrow \left[\, D^{k-1}; \phi_{m^k} \,\right]$
8: Coefficients actifs : $x^k \leftarrow \arg\min_x \left\| \, y - x\, D^k \, \right\|_2^2$
9: Résidu : $\epsilon^k \leftarrow y - x^k\, D^k$
10: $k \leftarrow k + 1$
11: **until** critère d'arrêt

Algorithm 9 : $x = \text{Q-OMPr}\ (y, \Phi)$

1: **initialisation** : $k = 1,\ \epsilon^0 = y,$ dictionnaire $D^0 = \varnothing$
2: **repeat**
3: **for** $m \leftarrow 1, M$ **do**
4: Produits scalaires : $C^k_m \leftarrow \left\langle\, \epsilon^{k-1}, \phi_m \,\right\rangle_{\mathsf{r}} = \phi^H_m\, \epsilon^{k-1}$
5: **end for**
6: Sélection : $m^k \leftarrow \arg\max_m \,\left|\, C^k_m \,\right|$
7: Dictionnaire actif : $D^k \leftarrow \left[\, D^{k-1}, \phi_{m^k} \,\right]$
8: Coefficients actifs : $x^k \leftarrow \arg\min_x \left\| \, y - D^k x \, \right\|_2^2$
9: Résidu : $\epsilon^k \leftarrow y - D^k x^k$
10: $k \leftarrow k + 1$
11: **until** critère d'arrêt

3.2.2 Extension des Q-OMP au cas d'invariance par translation

Dans le cadre de l'étude de signaux temps-série, nous spécifions les algorithmes précédents au cas d'invariance par translation. Pour le Q-OMPl, le produit scalaire entre le résidu ϵ^{k-1} et chaque atome ϕ_m (étape 4) est maintenant remplacé par la corrélation avec chaque noyau ψ_l. La corrélation quaternionique a été introduite dans [MSE03]. Dans le cas de multiplication à gauche, la corrélation quaternionique non-circulaire $\Gamma \in \mathbb{H}^{N_1+N_2-1}$ entre les signaux $\mathfrak{q}_1(t) \in \mathbb{H}^{1\times N_1}$ et $\mathfrak{q}_2(t) \in \mathbb{H}^{1\times N_2}$ est :

$$\Gamma_{\mathsf{l}}\left\{\mathfrak{q}_1, \mathfrak{q}_2\right\}(\tau) = \left\langle \mathfrak{q}_1(t), \mathfrak{q}_2(t-\tau) \right\rangle_{\mathsf{l}} = \mathfrak{q}_1(t-\tau)\, \mathfrak{q}_2^H(t) \,. \qquad (3.25)$$

La sélection (étape 6) détermine l'atome optimal qui est caractérisé par son indice de noyau l^k et sa position τ^k. La projection orthogonale (étape 8) donne le vecteur $x^k = \left[x_{l^1,\tau^1},\ x_{l^2,\tau^2}\ ...\ x_{l^k,\tau^k}\right]$. Finalement, l'Eq. (3.23) de l'approximation K-parcimonieuse devient :

$$\hat{y}^K = \sum_{k=1}^{K} x_{l^k,\tau^k}\ \psi_{l^k}(t-\tau^k)\ . \tag{3.26}$$

Pour le Q-OMPr qui a la multiplication à droite, le produit scalaire (étape 4) est remplacé par la corrélation quaternionique définie comme :

$$\Gamma_{\mathsf{r}}\left\{\mathfrak{q}_1,\mathfrak{q}_2\right\}(\tau) = \left\langle \mathfrak{q}_1(t), \mathfrak{q}_2(t-\tau)\right\rangle_{\mathsf{r}} = \mathfrak{q}_2^H(t-\tau)\ \mathfrak{q}_1(t)\ . \tag{3.27}$$

De même, l'Eq. (3.24) de l'approximation K-parcimonieuse devient :

$$\hat{y}^K = \sum_{k=1}^{K} \psi_{l^k}(t-\tau^k)\ x_{l^k,\tau^k}\ . \tag{3.28}$$

3.2.3 Spikegramme

Nous expliquons maintenant comment visualiser les coefficients obtenus à partir d'une décomposition quaternionique invariante par translation. Comme expliqué en Section 1.2.3, le spikegramme classique représente trois informations par coefficient x_{l^k,τ^k} actif :

- la position temporelle τ^k en abscisse,
- l'indice de noyaux l^k en ordonnée,
- l'amplitude du coefficient x_{l^k,τ^k} en échelle de couleur.

Dans le cas présent, les coefficients sont quaternioniques, ce qui fait que l'amplitude est représentée par quatre paramètres à afficher pour chaque coefficient. Pour effectuer cela tout en gardant une bonne visualisation, chaque coefficient quaternionique est écrit tel que :

$$x_{l,\tau} = |x_{l,\tau}| \cdot q_{\,l,\tau} \qquad \text{et} \qquad \mathfrak{q}_{\,l,\tau} = e^{\mathsf{i}\theta^1_{l,\tau}} \cdot e^{\mathsf{k}\theta^2_{l,\tau}} \cdot e^{\mathsf{j}\theta^3_{l,\tau}}\ , \tag{3.29}$$

avec $|x_{l,\tau}|$ le module du coefficient qui représente l'énergie de l'atome, et $\mathfrak{q}_{\,l,\tau}$ un quaternion unitaire (*i.e.* son module est égal à 1). Ce quaternion unitaire n'a que trois degrés de liberté, que nous définissons arbitrairement comme les angles d'Euler [Han06]. Ces paramètres décrivent de manière univoque le quaternion considéré sur la sphère unité. Par la suite, nous utiliserons ce formalisme d'angles, bien qu'il n'y ait pas de rotation dans les traitements.

Deux échelles de couleur sont utilisées pour le spikegramme quaternionique : l'une pour l'amplitude des coefficients et l'autre pour les autres paramètres assimilés aux angles d'Euler. L'échelle des angles, définie de -180 à +180 degrés, est visuellement circulaire ; une valeur négative juste au-dessus de -180 apparaît donc visuellement proche d'une valeur juste en dessous de +180. Au final, chaque coefficient quaternionique x_{l^k,τ^k} est affiché avec six indications :

- la position temporelle τ^k en abscisse,
- l'indice de noyaux l^k en ordonnée,
- l'amplitude du coefficient $\left|x_{l^k,\tau^k}\right|$ en échelle de couleur,
- les 3 paramètres $\theta^1_{l^k,\tau^k}$, $\theta^2_{l^k,\tau^k}$, $\theta^3_{l^k,\tau^k}$ affichés verticalement (échelle de couleur circulaire).

Cette représentation est utilisée dans les figures suivantes, et fournit une visualisation intuitive des différents paramètres.

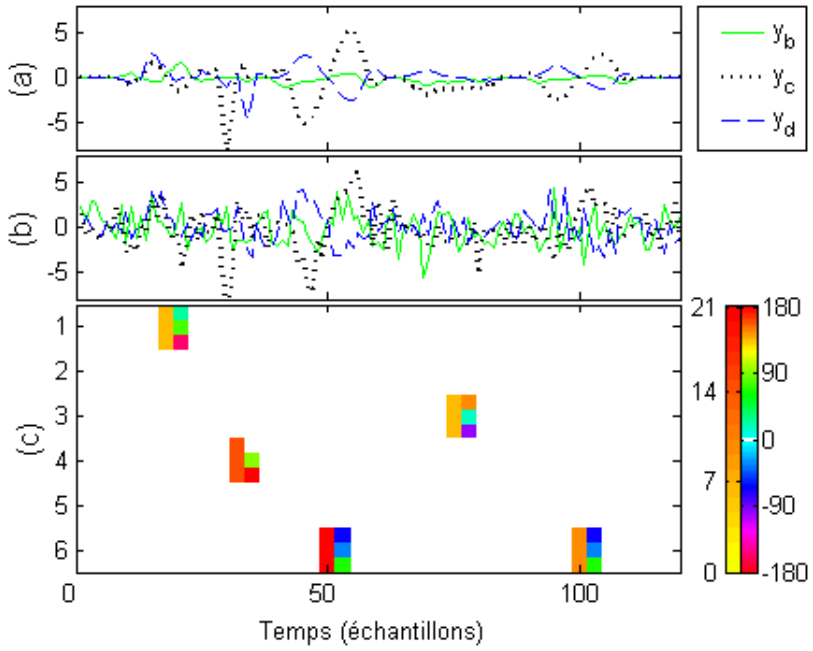


Fig. 3.1 – Signaux quaternioniques original (a) et bruité (b) (première partie imaginaire y_b représentée par un ligne verte pleine ; la seconde partie y_c par une ligne noire pointillée ; et la troisième partie y_d par une ligne traitillée bleue), et le spikegramme associé (c).

3.3 Expériences sur données simulées

Dans cette section, les algorithmes Q-OMPr et Q-OMPl sont illustrés dans un contexte de déconvolution de données simulées, et comparés au M-OMP.

3.3.1 Débruitage et déconvolution

Dans cette expérience de débruitage et déconvolution, les données sont trivariées (plutôt que quadrivariées) seulement pour ne pas surcharger les figures, et avoir une lecture claire. Les signaux trivariés, rentrés dans des quaternions purs, sont traités en utilisant des quaternions pleins comme coefficients. Un dictionnaire Ψ de $L = 6$ noyaux non-orthogonaux est construit artificiellement, et cinq coefficients $x_{l,\tau}$ sont générés (de telle sorte que les atomes se recouvrent). D'abord, le signal quaternionique $y \in \mathbb{H}^N$ est formé en utilisant l'Eq. (3.28) du modèle de multiplication à droite, et est affiché en Fig. 3.1(a). La première partie imaginaire y_b est représentée par une ligne verte, la seconde partie y_c par une ligne noire, et la troisième partie y_d par une ligne bleue. Ensuite, un bruit blanc Gaussien est ajouté, ce qui donne le signal bruité y_n maintenant caractérisé par un RSB de 0 dB. Il est affiché en Fig. 3.1(b), en gardant la même convention pour les styles de lignes. Les coefficients de synthèse sont affichés en Fig. 3.1(c) en utilisant le spikegramme introduit en Section 3.2.3.

Ensuite, nous déconvoluons ce signal y_n avec le dictionnaire Ψ en utilisant le Q-OMPr avec $K = 5$ itérations. Le signal débruité $\hat{y}_n$, qui est obtenu en calculant l'approximation K-parcimonieuse de y_n, est affiché en Fig. 3.2(a). Les coefficients $x_{l,\tau}$ sont le résultat de la déconvolution, et sont montrés en Fig. 3.2(b). En comparant les Fig. 3.1(c) et Fig. 3.2(b), nous observons que le Q-OMPr identifie

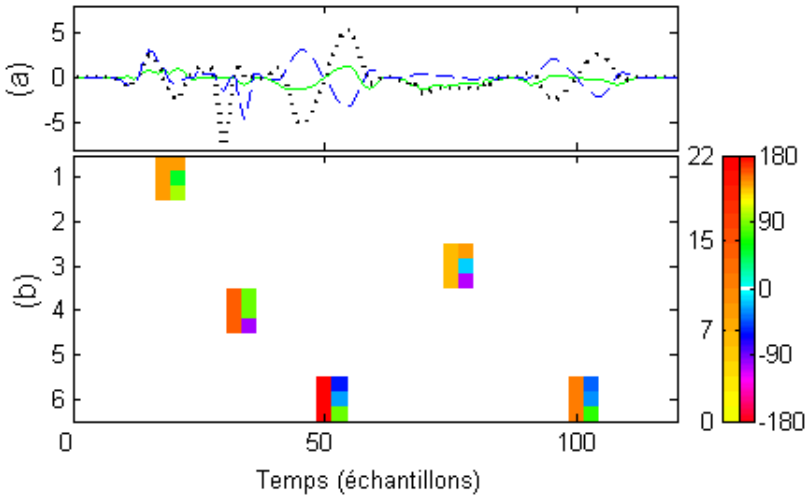


Fig. 3.2 – Signal quaternionique approximé par Q-OMPr (a) et le spikegramme associé (b).

bien les coefficients générés, et l'approximation $\hat{y}_n$ est proche du signal original y ; la rRMSE est de 20.1 %. Cette expérience est répétée 100 fois (avec les coefficients tirés aléatoirement), et la rRMSE moyenne est 22.1%. Ce résultat illustre l'efficacité du Q-OMPr pour le débruitage et la déconvolution de signaux quaternioniques. Remarquons qu'en Fig. 3.2(b), les coefficients $x_{6,50}$ et $x_{6,100}$ sont codés avec des amplitudes différentes mais avec le même quaternion unitaire $\mathfrak{q}$.

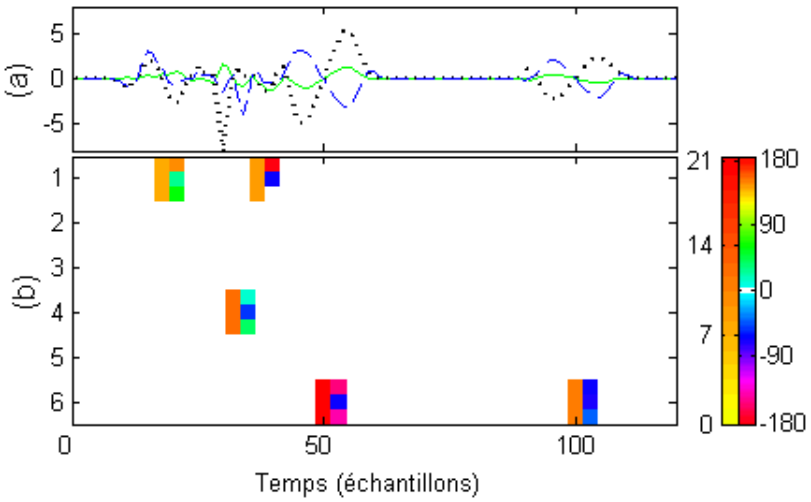


Fig. 3.3 – Signal quaternionique approximé par Q-OMPl (a) et le spikegramme associé (b).

Nous comparons ces résultats avec ceux obtenus avec le Q-OMPl. De la même manière, le Q-OMPl est utilisé avec $K = 5$ itérations, donnant le signal débruité en Fig. 3.3(a). La rRMSE est de 45.8%, et moyennée sur 100 expériences, 35.4%. Le spikegramme associé est montré en Fig. 3.3(b). Nous observons que les coefficients sont plutôt bien retrouvés, même s'il y a des erreurs dues au fait que le modèle multiplicatif n'est pas adapté aux données considérées.

Le M-OMP est maintenant comparé, utilisé dans un cas trivarié. Le signal quaternionique pur y_n

est maintenant rentré dans un signal réel trivarié $\underline{y}_n \in \mathbb{R}^{N\times 3}$, et il en est de même pour le dictionnaire de noyaux. Le M-OMP est appliqué avec $K = 5$ itérations, et donne le signal débruité $\underline{\hat{y}}_n$ affiché en Fig. 3.4(a). La rMSE est de 82.9%, et moyennée sur 100 expériences, 82.7%. Le spikegramme associé est montré en Fig. 3.4(b), en utilisant la visualisation classique (cf Section 1.2.3). Nous observons que les forts coefficients sont plutôt bien retrouvés, alors que les autres ne le sont pas (notamment au niveau du décalage temporel τ, de l'indice de noyau l, et de l'amplitude). Cependant, bien que les plus forts coefficients soient bien identifiés, cela n'est pas suffisant pour obtenir une approximation satisfaisante. En fait, l'approximation parcimonieuse multivariée n'est pas adaptée aux données de cette expérience, étant donné qu'elle ne prend pas en compte les termes croisés de la partie quaternionique vectorielle.

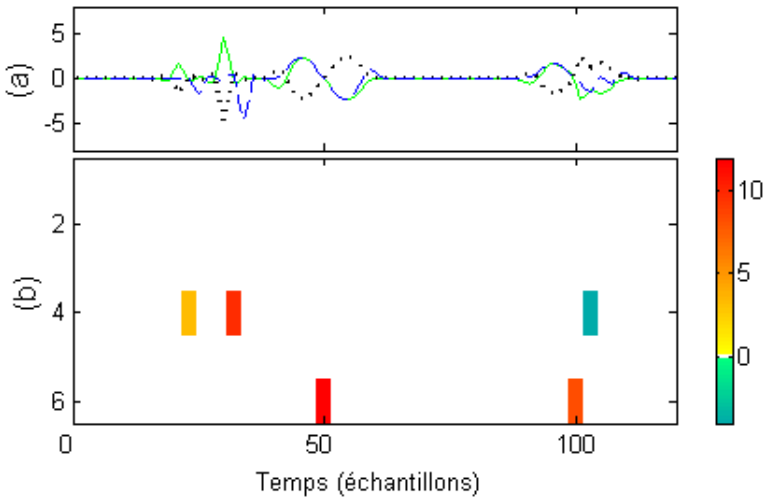


Fig. 3.4 – Signal réel trivarié $\underline{\hat{y}}_n$ approximé par M-OMP (a) et le spikegramme associé (b).

3.3.2 Comparaisons

Nous comparons maintenant les trois algorithmes dans des conditions plus générales. Nous étudions 100 signaux quaternioniques pleins, composés de $N = 256$ échantillons. Comme dans l'expérience précédente, les signaux sont générés comme une combinaison linéaire de $K = 15$ atomes, avec des coefficients tirés aléatoirement, mais sans bruit. Nous effectuons l'approximation K-parcimonieuse des signaux, et nous relevons la rRMSE en fonction des itérations internes des algorithmes. Cette erreur est ensuite moyennée sur les 100 signaux. Nous distinguons deux cas : le premier où les données sont générées avec un modèle linéaire avec multiplication à droite, et les résultats sont affichés en Fig. 3.5 ; et le second avec un modèle linéaire avec multiplication à gauche, et les résultats sont affichés en Fig. 3.6.

Nous constatons que le Q-OMPr (*resp.* Q-OMPl) donne de meilleurs résultats pour les signaux générés avec le modèle multiplicatif à droite en Fig. 3.5 (*resp.* à gauche en Fig. 3.6). Cela prouve bien la nécessité d'avoir deux algorithmes d'approximation parcominieuse. Le M-OMP se comporte de façon identique dans les deux cas. Le Q-OMPr (*resp.* Q-OMPl) n'est pas exactement à zéro pour $K = 15$ sur la Fig. 3.5 (*resp.* Fig. 3.6) : ceci est dû à la non-convexité de la poursuite ℓ_0 qui peut tomber dans un minimum local en cas de fort recouvrement d'atomes.

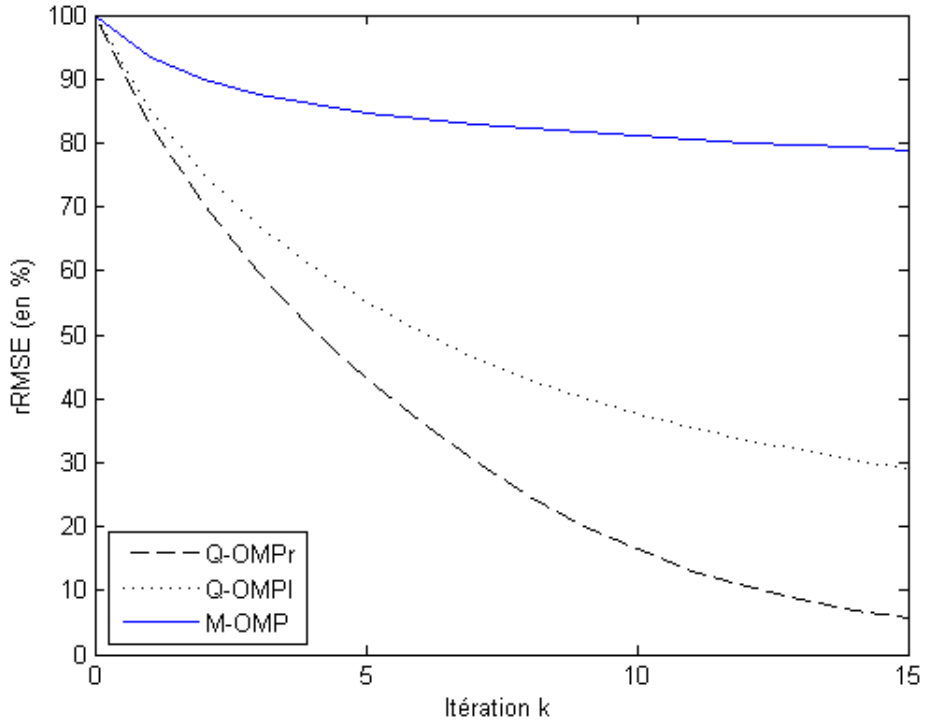


Fig. 3.5 – rRMSE moyennée pour le Q-OMPr, le Q-OMPl et le M-OMP sur des signaux simulés avec multiplication à droite, en fonction de l'itération interne k.

3.3.3 Complexité

Les algorithmes Q-OMPr et Q-OMPl ont les mêmes complexités, car seul l'ordre des variables diffère lors des étapes de calcul. Comme constaté dans l'expérience précédente, chacun des deux est adapté au modèle multiplicatif lui correspondant. Par la suite, le Q-OMPr et le Q-OMPl seront rassemblés sous le nom Q-OMP.

Maintenant, nous considérons un signal quaternionique plein $y \in \mathbb{H}^N$ donnant un signal réel quadrivarié $\boldsymbol{y} \in \mathbb{R}^{N\times 4}$. Si les coefficients sont strictement réels, les méthodes M-OMP et Q-OMP sont équivalentes ; sinon, le Q-OMP donne de meilleures performances selon son modèle linéaire qui lui est propre. D'un point de vue complexité, la corrélation quadrivariée possède $V = 4$ termes, alors que celle quaternionique en a 16 : le Q-OMP a donc une complexité multipliée par 4.

Conclusion

Dans ce chapitre, nous avons présenté deux méthodes d'approximation parcimonieuse pour des signaux quaternioniques. A cause de la non-commutativité, deux modèles sont considérés : le Q-OMPr est dédié au modèle de multiplication à droite, et le Q-OMPl à gauche.

Les applications des Q-OMP sont les traitements quaternioniques comme la déconvolution, le débruitage, la sélection de variable et tous les traitements habituels utilisant la parcimonie. Une des pistes à explorer consiste à mettre en place un *Quaternionic* DLA (Q-DLA) en couplant le Q-OMP avec une étape de mise à jour du dictionnaire par descente de gradient. L'objectif est de fournir un dictionnaire

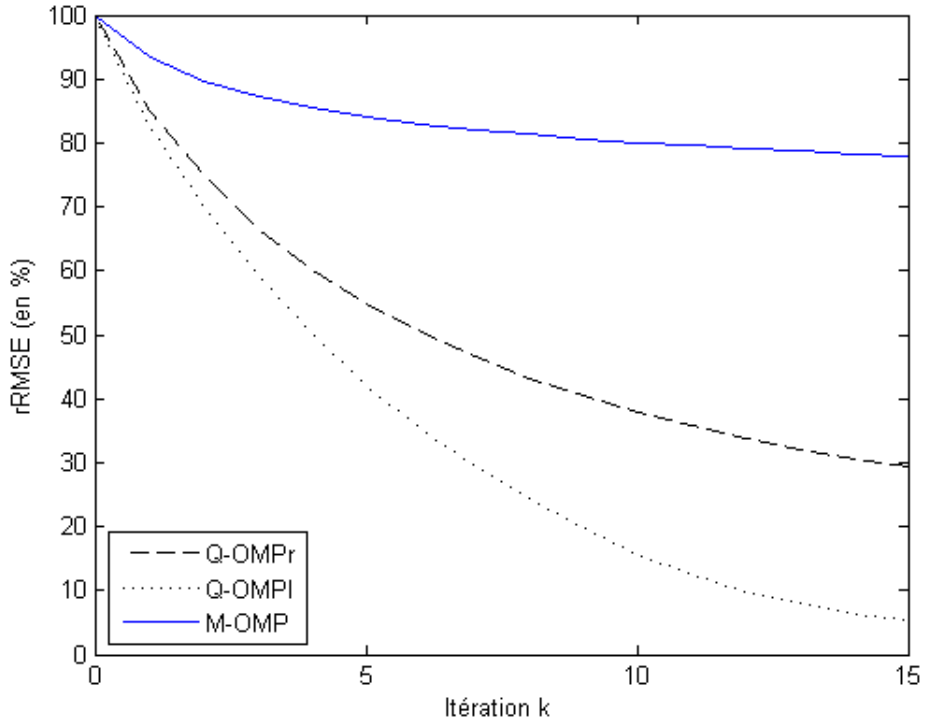


Fig. 3.6 – rRMSE moyennée pour le Q-OMPr, le Q-OMPl et le M-OMP sur des signaux simulés avec multiplication à gauche, en fonction de l'itération interne k.

quaternionique adapté afin d'améliorer les résultats par rapport à des dictionnaires classiques, comme cela a été largement constaté dans le cas réel classique. Par exemple, dans le cas du débruitage d'images couleur [AC12], le dictionnaire analytique utilise les paquets d'ondelettes basés sur les polynômes quaternioniques.

Bibliographie

[AC12] B. Augereau et P. Carré : Hypercomplex polynomial wavelet packet application for color image. *In Conf. Applied Geometric Algebras in Computer Science and Engineering AGACSE*, 2012.

[ES07] T.A. Ell et S.J. Sangwine : Hypercomplex Fourier transforms of color images. *IEEE Trans. on Image Processing*, 16:22–35, 2007.

[Han06] A.J. Hanson : *Visualizing quaternions*. Morgan-Kaufmann/Elsevier, 2006.

[JTJ+11] S. Javidi, C.C. Took, C. Jahanchahi, N. Le Bihan et D.P. Mandic : Blind extraction of improper quaternion sources. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '11*, pages 3708–3711, 2011.

[MLBM06] S. Miron, N. Le Bihan et J.I. Mars : Quaternion-MUSIC for vector-sensor array processing. *IEEE Trans. on Signal Processing*, 54:1218–1229, 2006.

[MSE03] C.E. Moxey, S.J. Sangwine et T.A. Ell : Hypercomplex correlation techniques for vector images. *IEEE Trans. on Signal Processing*, 51:1941–1953, 2003.

[PRK93] Y.C. Pati, R. Rezaiifar et P.S. Krishnaprasad : Orthogonal Matching Pursuit : recursive function approximation with applications to wavelet decomposition. *In Conf. Record of the Asilomar Conf. on Signals, Systems and Comput.*, pages 40–44, 1993.

[TM10] C.C. Took et D.P. Mandic : Quaternion-valued stochastic gradient-based adaptive IIR filtering. *IEEE Trans. on Signal Processing*, 58:3895–3901, 2010.

[Zha97] F. Zhang : Quaternions and matrices of quaternions. *Linear Algebra and its Applications*, 251:21–57, 1997.

Chapitre 4

Représentations invariantes par rotation 2D

Introduction

Dans les chapitres précédents, nous nous sommes intéressés aux signaux multivariés et au cas particulier des signaux quaternioniques qui permettent de traiter des signaux de dimension 3 ou 4 avec un modèle spécifique. Dans ce chapitre, nous nous restreindrons à des signaux réels bivariés mais pour lesquels un degré de liberté supplémentaire est inclus avec l'invariance par rotation 2D (en anglais, *2D rotation invariant* (2DRI)). L'application pour illustrer cette approche est naturellement le mouvement 2D pour lequel nous souhaitons que la représentation soit indépendante de l'orientation de l'exécution du geste dans l'espace 2D.

Ayant présenté le M-OMP et le M-DLA, ces méthodes sont spécifiées dans le cas de l'invariance par rotation 2D en ajoutant un degré de liberté. Elles seront illustrées sur des données réelles d'écriture manuscrite.

4.1 Invariance par rotation 2D

Dans cette section, nous expliquons comment intégrer dans le modèle multivarié précédemment évoqué un degré de liberté supplémentaire pour obtenir l'invariance par rotation. Nous ferons aussi un rapide état de l'art sur les méthodes d'apprentissage de mouvement.

4.1.1 Formulation du problème

Nous traitons désormais des données réelles bivariées, *i.e.* avec $V = 2$, et le signal désormais étudié est $\boldsymbol{y} \in \mathbb{R}^{N\times 2}$. L'Eq. (2.4) de décomposition du modèle multivarié devient :

$$\begin{pmatrix} \boldsymbol{y}[1](t) \\ \boldsymbol{y}[2](t) \end{pmatrix} = \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} x_{l,\tau} \begin{pmatrix} \boldsymbol{\psi}_l[1](t-\tau) \\ \boldsymbol{\psi}_l[2](t-\tau) \end{pmatrix} + \begin{pmatrix} \boldsymbol{\epsilon}[1](t) \\ \boldsymbol{\epsilon}[2](t) \end{pmatrix}, \tag{4.1}$$

avec $\begin{pmatrix} \end{pmatrix}$ représentant la concaténation multivariée, *i.e.* selon la $3^{\text{ième}}$ dimension. Ce cas sera appelé *orienté* par la suite, étant donné que les atomes bivariés ne peuvent pas tourner, et qu'ils sont définis

dans une orientation fixée.

En étudiant des signaux bivariés tels que des mouvements 2D, nous voulons que leur représentation soit indépendante de leurs orientations. Quelle que soit l'orientation dans laquelle est réalisé le mouvement, nous souhaitons une décomposition identique : la répresentation est robuste à la rotation, et nous diminuons la redondance du dictionnaire qu'il faudrait apprendre pour chacune des orientations possibles. L'invariance par rotation implique l'introduction d'une matrice de rotation $R \in \mathbb{R}^{2\times 2}$, d'angle $\theta_{l,\tau}$, pour chaque atome bivarié $\boldsymbol{\psi}_l(t-\tau)$. Ainsi, l'Eq. (4.1) devient :

$$\begin{pmatrix} \boldsymbol{y}[1](t) \\ \boldsymbol{y}[2](t) \end{pmatrix} = \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} x_{l,\tau}\, R(\theta_{l,\tau}) \begin{pmatrix} \boldsymbol{\psi}_l[1](t-\tau) \\ \boldsymbol{\psi}_l[2](t-\tau) \end{pmatrix} + \begin{pmatrix} \boldsymbol{\epsilon}[1](t) \\ \boldsymbol{\epsilon}[2](t) \end{pmatrix} . \tag{4.2}$$

L'approximation parcimonieuse multivariée (2.12) spécifiée au cas de l'invariance par rotation 2D devient :

$$\min_{x,\theta} \left\| \begin{pmatrix} \boldsymbol{y}[1](t) \\ \boldsymbol{y}[2](t) \end{pmatrix} - \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} x_{l,\tau}\, R(\theta_{l,\tau}) \begin{pmatrix} \boldsymbol{\psi}_l[1](t-\tau) \\ \boldsymbol{\psi}_l[2](t-\tau) \end{pmatrix} \right\|_F^2$$

$$\text{t.q. } \|x\|_0 \leq K \quad \text{et} \quad \forall l \in \mathbb{N}_L, \forall \tau \in \sigma_l,\ R(\theta_{l,\tau}) R(\theta_{l,\tau})^T = Id\,. \tag{4.3}$$

Il nous faut désormais estimer les coefficients x et les angles θ.

Dans le cas multicapteurs, nous considérons P capteurs faisant l'acquisition de données bivariées. Les capteurs sont physiquement liés, et sont donc soumis à la même rotation. Par exemple, dans un repère spatial cartésien (x, y), nous étudions des signaux bivariés issus d'un capteur de vitesse (donnant les vitesses $\mathfrak{v}_\mathsf{x}$ et $\mathfrak{v}_\mathsf{y}$), d'un accéléromètre (donnant les accélérations $\mathfrak{a}_\mathsf{x}$ et $\mathfrak{a}_\mathsf{y}$), d'un gyromètre (donnant les vitesses angulaires $\mathfrak{g}_\mathsf{x}$ et $\mathfrak{g}_\mathsf{y}$), etc. Nous souhaitons aussi estimer les coefficients et les angles dans ce cas multicapteurs.

4.1.2 Méthodes d'apprentissage de primitives du mouvement

Dans ce paragraphe, nous faisons un rapide état de l'art concernant les méthodes qui apprennent des primitives, *i.e.* atomes, du mouvement. Plusieurs communautés telles que la robotique, les neurosciences ou les télécommunications, sont intéressées par l'apprentissage des primitives spatio-temporelles spécifiques à des tâches [SSAS11].

Les données *Character Trajectories* que nous traiterons dans ce chapitre ont été initialement traitées avec un modèle de Markov caché factoriel et une méthode d'apprentissage EM [WTS07, WTS08]. Dans [KSU10], Kim *et al.* utilisent une factorisation parcimonieuse de tenseur avec des contraintes de normes mixtes tensorielles pour effectuer un apprentissage multicomposante. Dans [MTSS11, MTS12], Meier *et al.* modélisent les mouvements par un système dynamique, et un algorithme EM leur permet d'apprendre une librairie de primitives. Dans [DL12], Dai et Lücke utilisent une approche EM variationnelle pour apprendre des primitives du mouvement dans un cas de données comportant des occlusions. Dans [VEG12], Vollmer *et al.* utilisent une factorisation en matrice non-négative qui intègre l'invariance par translation, mais le recouvrement de primitives est pénalisé.

Cependant, aucune de ces méthodes d'extraction de primitives ne traite du problème d'invariance par rotation.

4.2 Méthodes 2DRI

Dans cette section, nous présentons l'algorithme d'approximation parcimonieuse 2DRI-OMP qui résout le problème (4.3) en utilisant le principe de l'OMP.

4.2.1 Approche 2DRI

Nous cherchons à adapter le M-OMP (Algorithme 6), qui traite les signaux multivariés, au cas 2DRI pour résoudre le problème présenté en Eq. (4.3). Désormais, dans l'étape de sélection (Algorithme 6, étape 6), le but sera de sélectionner l'atome optimal dans son orientation optimale, et donc de trouver son angle θ_{l^k,τ^k} qui maximise la corrélation $\left|C_l^k(\tau,\theta_{l,\tau})\right|$. Une approche naïve consiste à échantillonner la variable $\theta_{l,\tau}$ en Θ angles et d'ajouter un nouveau degré de liberté dans le calcul des corrélations (Algorithme 6, étape 4). La complexité du calcul est augmentée d'un facteur Θ par rapport au M-OMP utilisé dans le cas orienté. Notons que cette idée est utilisée pour traiter des signaux bidimensionnels $\boldsymbol{y} \in \mathbb{R}^{N_1 \times N_2}$ tels que des images [WEK03, MS11], mais ce problème est différent du nôtre.

Pour éviter d'ajouter ce coût, nous transformons le signal réel bivarié $\boldsymbol{y} \in \mathbb{R}^{N\times 2}$ en un signal complexe $y \in \mathbb{C}^N$:

$$y \leftarrow \boldsymbol{y}[1] + \boldsymbol{y}[2]\mathsf{i}\,, \tag{4.4}$$

avec i le nombre complexe imaginaire. Les noyaux et les coefficients sont eux aussi complexes. Retrouvant l'Eq. (1.4) dans un cas complexe, nous appliquons désormais le M-OMP spécifié au cas complexe (cf. Section 4.2.2) où les coefficients complexes codent l'amplitude grâce au module et l'angle de rotation grâce à l'argument :

$$x_{l,\tau} = |x_{l,\tau}| \cdot e^{\,\mathsf{i}\,\theta_{l,\tau}}\,. \tag{4.5}$$

Ainsi, le modèle (4.2) est transformé en complexe grâce aux Eq. (4.4) et (4.5), et au final, la décomposition du signal $y \in \mathbb{C}^N$ s'écrit :

$$y(t) = \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} |x_{l,\tau}| \cdot e^{\,\mathsf{i}\,\theta_{l,\tau}} \cdot \psi_l(t-\tau) + \epsilon(t)\,. \tag{4.6}$$

Grâce au degré de liberté de rotation ajouté, les noyaux peuvent maintenant tourner puisqu'ils ne sont plus figés dans une orientation spécifique. Les noyaux sont invariants par translation et par rotation, produisant une décomposition *non-orientée* (M-OMP avec $V = 1$ et $y \in \mathbb{C}^N$), contrairement à l'approche précédente appelée *orientée* (M-OMP avec $V = 2$ et $\boldsymbol{y} \in \mathbb{R}^{N\times 2}$).

Dans le cas multicapteurs, ces $\mathsf{P} = 3$ signaux (vitesse, accélération et vitesse angulaire) peuvent être agrégés ensemble dans un signal $\boldsymbol{y} \in \mathbb{C}^{N\times \mathsf{P}}$ tel que :

$$\boldsymbol{y} \leftarrow \begin{pmatrix} \mathfrak{v}_\mathsf{x} + \mathfrak{v}_\mathsf{y}\mathsf{i} \\ \mathfrak{a}_\mathsf{x} + \mathfrak{a}_\mathsf{y}\mathsf{i} \\ \mathfrak{g}_\mathsf{x} + \mathfrak{g}_\mathsf{y}\mathsf{i} \end{pmatrix}\,. \tag{4.7}$$

Avec P capteurs, la décomposition non-orientée (4.6) devient :

$$\boldsymbol{y}(t) = \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} |x_{l,\tau}| \cdot e^{\,\mathsf{i}\,\theta_{l,\tau}} \cdot \boldsymbol{\psi}_l(t-\tau) + \boldsymbol{\epsilon}(t)\,. \tag{4.8}$$

L'angle de la rotation commune est choisi conjointement entre les P composantes complexes. Il en résulte donc une décomposition multicapteurs non-orientée (M-OMP avec $V = \mathsf{P}$ et $\boldsymbol{y} \in \mathbb{C}^{N \times \mathsf{P}}$). Si les P capteurs ne sont pas soumis à la même rotation, les signaux doivent être traités par un autre moyen (cf. Perspectives).

Cette spécification au cas 2DRI de l'approximation parcimonieuse M-OMP (*resp.* apprentissage de dictionnaires M-DLA) est maintenant appelée 2DRI-OMP (*resp.* 2DRI-DLA). Nous détaillons maintenant le 2DRI-OMP et le 2DRI-DLA qui analysent des signaux complexes.

4.2.2 Description du 2DRI-OMP et du 2DRI-DLA

Pour traiter des signaux complexes, nous définissons le produit scalaire hermitien comme $\langle A, B \rangle = \mathrm{Tr}(B^H A)$, avec $(.)^H$ représentant l'opérateur transconjugué.

Les critères utilisés en Section 2.2 pour le M-OMP (Algorithme 6) doivent être reconsidérés dans un cadre complexe. Nous rappelons qu'à l'itération courante k, le 2DRI-OMP sélectionne l'atome qui produit la plus forte décroissance (en valeur absolue) de l'erreur quadratique moyenne $\left\|\boldsymbol{\epsilon}^{k-1}\right\|_F^2 = \mathrm{Tr}({\boldsymbol{\epsilon}^{k-1}}^H \boldsymbol{\epsilon}^{k-1})$. L'opérateur gradient complexe est introduit par Brandwood [Bra83]. Considérant $z \in \mathbb{C}$, les règles de dérivation complexe sont :

$$\frac{\partial z^*}{\partial z} = \frac{\partial z}{\partial z^*} = 0 \quad \text{et} \quad \frac{\partial z}{\partial z} = \frac{\partial z^*}{\partial z^*} = 1 \,. \tag{4.9}$$

De plus, il montre que la direction de plus forte variation d'une fonction de coût à valeurs réelles $J(z)$ par rapport à z est $\partial J / \partial z^*$ [Bra83]. En notant $\boldsymbol{\epsilon}^{k-1} = x_m \boldsymbol{\phi}_m + \boldsymbol{\epsilon}^k$, nous avons :

$$\frac{\partial \left\|\boldsymbol{\epsilon}^{k-1}\right\|_F^2}{\partial x_m^*} = \mathrm{Tr}(\, {\boldsymbol{\phi}_m}^H \, \boldsymbol{\epsilon}^{k-1} \,) = \left\langle \boldsymbol{\epsilon}^{k-1}, \boldsymbol{\phi}_m \right\rangle \,. \tag{4.10}$$

Ainsi, l'atome produisant la plus forte décroissance de l'erreur est équivalent à l'atome ayant le plus fort produit scalaire complexe. Dans le cas d'invariance par translation, la corrélation complexe non-circulaire $\Gamma \in \mathbb{C}^{N_1+N_2-1}$ entre les signaux $\boldsymbol{y}_1(t) \in \mathbb{C}^{N_1 \times \mathsf{P}}$ et $\boldsymbol{y}_2(t) \in \mathbb{C}^{N_2 \times \mathsf{P}}$ est définie par :

$$\Gamma\left\{\boldsymbol{y}_1, \boldsymbol{y}_2\right\}(\tau) = \left\langle \boldsymbol{y}_1(t), \boldsymbol{y}_2(t-\tau) \right\rangle \,. \tag{4.11}$$

Le M-OMP est donc simplement étendu aux complexes avec la nouvelle définition du produit scalaire pour donner le 2DRI-OMP.

Le M-DLA (Algorithme 7) est lui aussi étendu aux signaux complexes pour donner le 2DRI-DLA : il itère entre le 2DRI-OMP et une mise à jour du dictionnaire adaptée aux complexes. En dérivant le critère par rapport à $\boldsymbol{\psi}_l^*$, nous obtenons la mise à jour des noyaux :

$$\boldsymbol{\psi}_l^i(\underline{t}) = \boldsymbol{\psi}_l^{i-1}(\underline{t}) + (H_l^i + \lambda^i . I)^{-1} \cdot \sum_{\tau \in \sigma_l} x_{l,\tau;p}^{i\,*} \, \boldsymbol{\epsilon}_p^{i-1}(\underline{t} + \tau) \,. \tag{4.12}$$

La différence avec l'Eq. (2.21) de mise à jour du M-DLA réside dans le fait que le coefficient $x_{l,\tau;p}^i$ est désormais conjugué. Dans le 2DRI-DLA, l'initialisation du dictionnaire se fait avec des signaux complexes (aléatoires ou issus de l'ensemble d'apprentissage).

4.2.3 Remarques sur le 2DRI-OMP

Notons que si le nombre d'atomes actifs est $K = 1$, le problème 2DRI considéré est identique au *2D curve matching* [SS85]. Schwartz et Sharir fournissent une solution analytique pour calculer $R(\theta_{l,\tau})$, mais cette approche est assez longue puisque le calcul est effectué pour chaque noyau l et pour chaque échantillon τ. L'utilisation de signaux complexes comme indiqué précédemment permet de résoudre ce problème facilement.

Considérant encore $K = 1$, Vlachos *et al.* [VGD04] fournissent des signatures invariantes par rotation 2D pour faire de la reconnaissance de trajectoires. Cependant, comme dans la plupart des méthodes utilisant des descripteurs invariants, leur méthode perd le paramètre de rotation, contrairement à notre approche.

4.3 Application aux données d'écriture manuscrite

Après avoir décrit les modèles et les méthodes 2DRI, nous allons présenter leurs applications à l'analyse de signaux de mouvement 2D.

4.3.1 Données de mouvements 2D

Nos méthodes sont appliquées aux signaux de mouvement *Character Trajectories* qui sont disponibles sur la base de données de University of California at Irvine (UCI) [FA10]. Ces données comportent 2858 lettres manuscrites qui sont acquises avec une tablette Wacom échantillonnée à 200 Hz, avec une centaine d'occurrences de 20 lettres [1] écrites par la même personne. Les signaux temporels sont les vitesses cartésiennes $\mathfrak{v}_x$ et $\mathfrak{v}_y$ de la pointe du stylo, ainsi que la dérivée de la pression, non considérée dans cette expérience. Une occurrence de chaque lettre de la base de données est représentée en Fig. 4.1. Comme les unités de vitesse ne sont pas précisées dans la description de la base de données, nous ne pouvons pas les définir pour cette expérience.

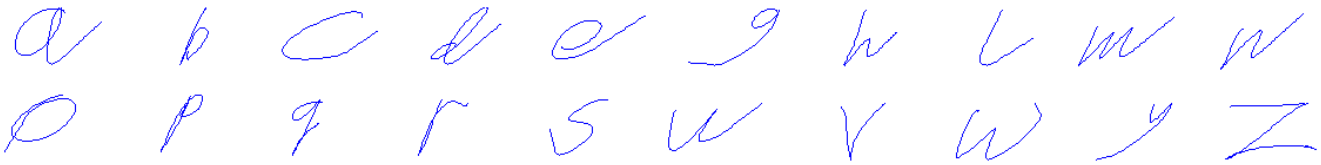


Fig. 4.1 – Visualisation d'une occurrence de chaque lettre de la base de données UCI *Character Trajectories*.

En utilisant les données brutes, nous souhaitons apprendre un dictionnaire adapté afin de représenter avec parcimonie les signaux de vitesse. La base de données est divisée en deux : un ensemble d'apprentissage, sur lequel sont appliqués les apprentissages de dictionnaire, qui est composé de 20 occurrences de chaque lettre (soit $P = 400$ caractères), et un ensemble de test pour pouvoir qualifier l'efficacité des représentations parcimonieuses ($Q = 2458$ caractères).

Les expériences sont faites dans le cas orienté avec le M-DLA et dans le cas non-orienté avec le 2DRI-DLA. Nous rappelons que l'agencement des données varie : dans le cas orienté, les signaux sont $\boldsymbol{y} \leftarrow$

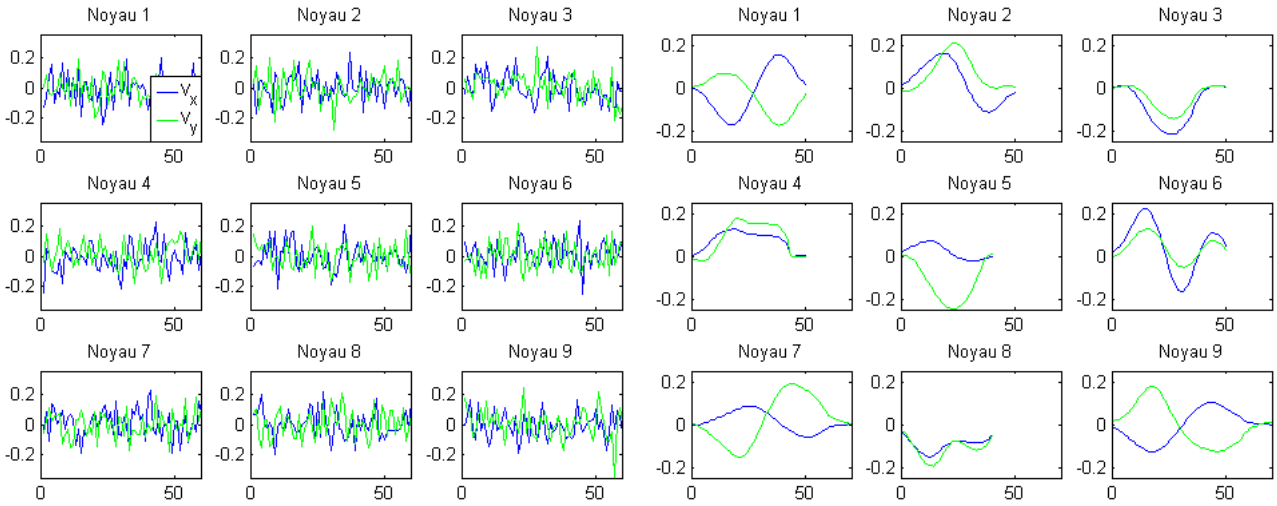
1. Les 6 lettres où le stylo est levé ne sont pas considérées.

$\left(\mathfrak{v}_x \, ; \, \mathfrak{v}_y \right)$, et dans le cas non-orienté $y \leftarrow \mathfrak{v}_x + \mathfrak{v}_y \mathrm{i}$. Dans ces deux cas, les algorithmes d'apprentissage sont initialisés sur des noyaux de bruit blanc Gaussien.

Trois expériences sont maintenant détaillées : l'apprentissage des dictionnaires, les décompositions sur les données, puis les décompositions sur les données tournées.

4.3.2 Expérience 1 : apprentissage de dictionnaires

Dans cette expérience, le 2DRI-DLA va fournir un dictionnaire appris non-orienté, soit en anglais *non-oriented learned dictionary* (NOLD). Les vitesses sont utilisées afin d'avoir des noyaux à bords nuls. Cela évite l'introduction de discontinuités dans le signal durant l'approximation parcimonieuse [2]. Le dictionnaire de noyaux est initialisé sur du bruit blanc Gaussien en Fig. 4.2(a), et le 2DRI-DLA est appliqué sur l'ensemble d'apprentissage. Nous obtenons un dictionnaire de noyaux de vitesse montré en Fig. 4.2(b), où chaque noyau est composé d'une partie réelle $\mathfrak{v}_x$ (bleu) et d'une partie imaginaire $\mathfrak{v}_y$ (vert). Cette convention pour le style des légendes de la Fig. 4.2 sera conservée tout au long du chapitre.



(a) Début de l'apprentissage.

(b) Fin de l'apprentissage.

Fig. 4.2 – Apprentissage d'un dictionnaire non-orienté : dictionnaire initialisé sur du bruit blanc Gaussien (a) et après apprentissage par 2DRI-DLA (b). Chaque noyau est composé d'une partie réelle $\mathfrak{v}_x$ (bleu) et d'une partie imaginaire $\mathfrak{v}_y$ (vert).

Les signaux de vitesse sont intégrés seulement pour fournir une représentation plus visuelle des noyaux. Cependant, à cause de l'intégration, deux noyaux de vitesse différents peuvent fournir des trajectoires (noyaux intégrés) très proches. Le dictionnaire de noyaux intégrés (Fig. 4.3) montre que des primitives du mouvement sont extraites avec succès grâce au 2DRI-DLA. En fait, les lignes droites (noyaux $l = 4$ et 8 par exemple) et courbées (noyaux $l = 1$ et 2 par exemple) de la Fig. 4.3 correspondent aux motifs élémentaires de l'ensemble des signaux manuscrits. Les étoiles marquent le point de départ des trajectoires.

La question est : comment choisir l'hyper-paramètre L correspondant à la taille du dictionnaire de

2. Remarquons aussi que contrairement aux signaux de positions, les signaux de vitesse permettent l'invariance spatiale (différente de l'invariance par translation temporelle).

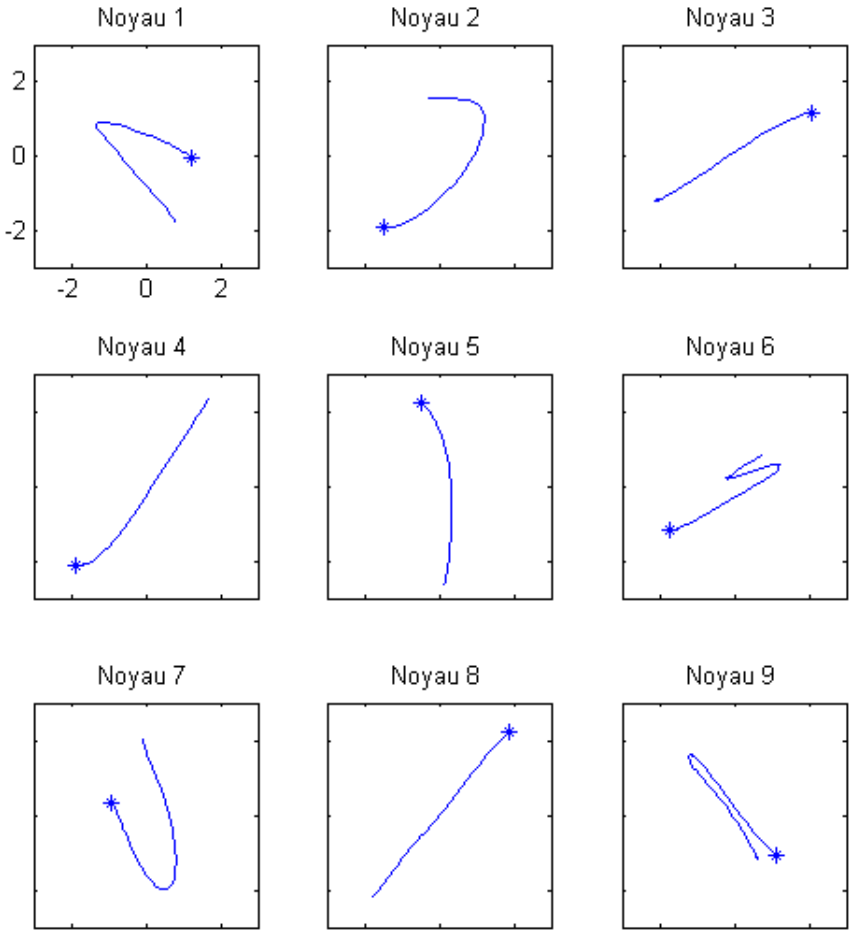


Fig. 4.3 – Dictionnaire de trajectoires associé au dictionnaire NOLD appris par le 2DRI-DLA.

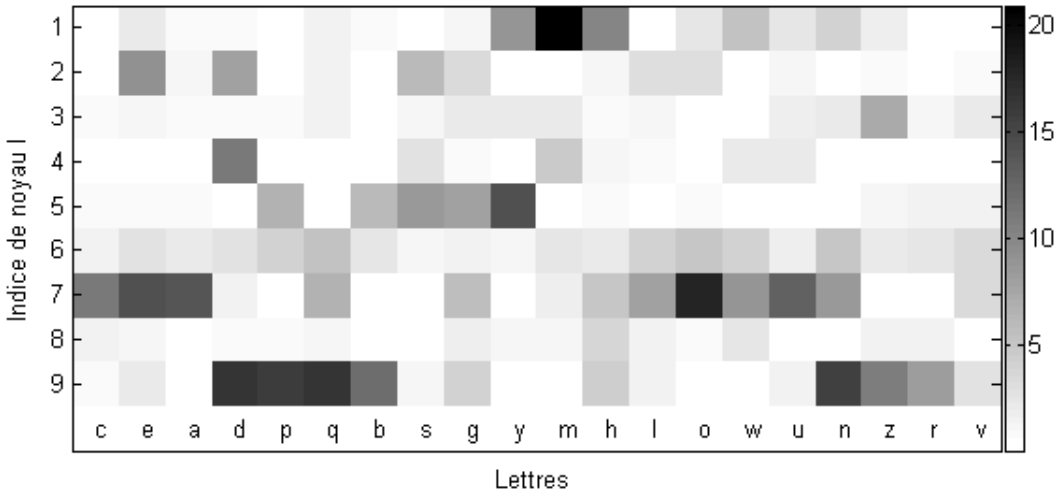


Fig. 4.4 – Matrice d'utilisation du dictionnaire calculée sur l'ensemble d'apprentissage. La moyenne des valeurs absolues des coefficients est donnée en fonction de l'indice de noyau l et de la lettre des signaux.

noyaux ? Dans le cas non-orienté, 9 noyaux sont utilisés alors que dans le cas orienté, ce nombre passe à 12. Ce choix est un compromis empirique entre la rRMSE finale obtenue sur l'ensemble d'apprentissage, la parcimonie du dictionnaire (petite taille), et l'interprétabilité du dictionnaire résultant (d'autres critères dépendant de l'application peuvent être aussi pris en compte).

Comme l'interprétabilité est un critère subjectif, une matrice d'utilisation est mise en place dans un cas supervisé (Fig. 4.4). La moyenne des valeurs absolues des coefficients (niveau de gris) calculée sur l'ensemble d'apprentissage est affichée en fonction de l'indice du noyau l (ordonnée) et de la classe du signal, *i.e.* la lettre (abscisse). Les lettres sont ordonnées selon les similarités de leurs profils d'utilisation. Nous pouvons dire qu'un dictionnaire a une bonne interprétabilité quand des noyaux très utilisés sont communs à des lettres différentes qui ont des parties communes. Par exemple, les lettres c, e et a ont des similarités de forme et partagent le noyau $l = 7$. De même, d et p partagent le noyau $l = 9$.

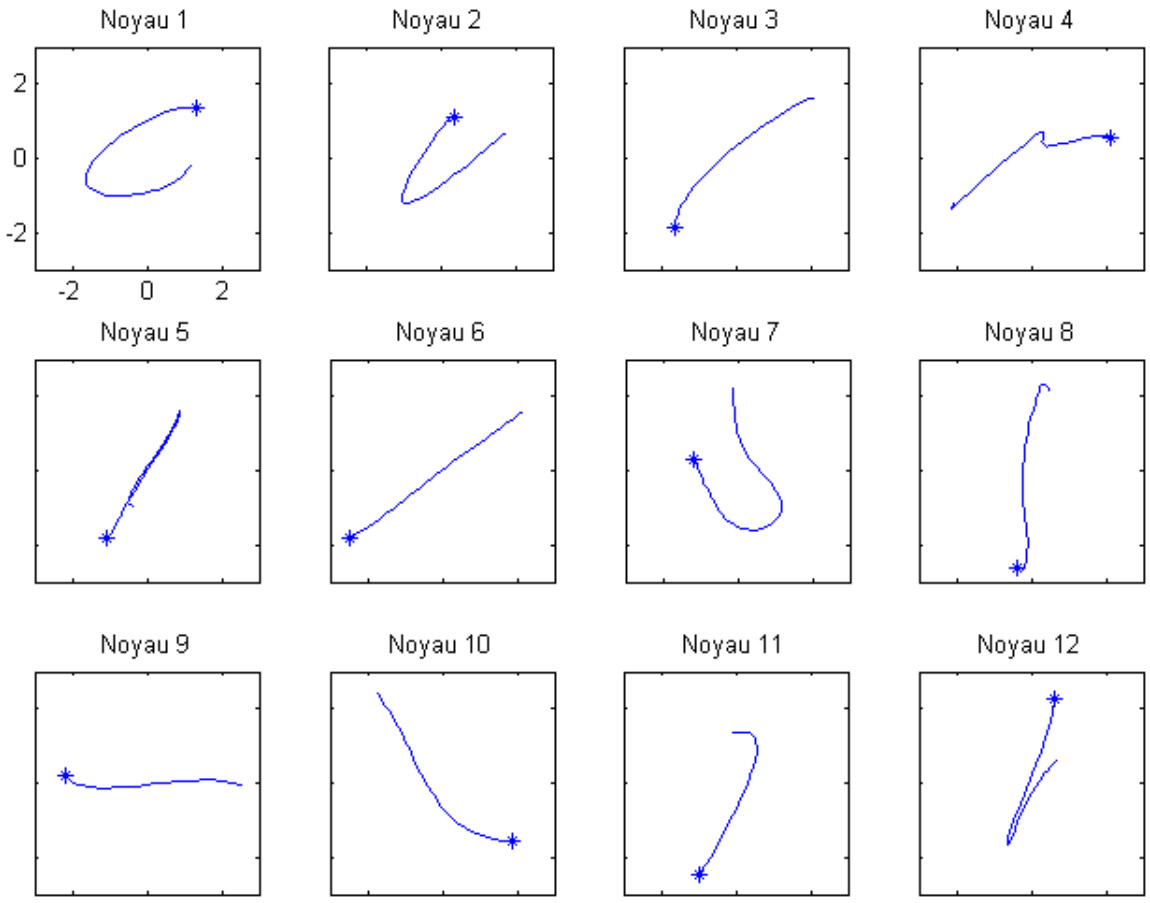


Fig. 4.5 – Dictionnaire de trajectoires associé au dictionnaire OLD appris par le M-DLA.

Nous remarquons aussi que durant le 2DRI-DLA / M-DLA, le 2DRI-OMP / M-OMP fournit une approximation K-parcimonieuse (cf. Sections 2.3 et 4.2.2). K est le nombre d'atomes actifs, et il détermine le nombre de primitives sous-jacentes qui sont cherchées dans chaque signal et ensuite apprises. Si la taille du dictionnaire L est trop petite comparée au nombre idéal de primitives que nous cherchons, les noyaux ne seront pas caractéristiques d'une forme particulière et la rRMSE sera élevée. A l'inverse, si L et K sont élevés, l'apprentissage de dictionnaire à tendance à éparpiller l'information dans les multiples noyaux.

Dans ce cas, la matrice d'utilisation sera très lisse, sans noyau caractéristique de lettres particulières. Si L est élevé et K est optimal, nous pouvons voir que certains noyaux seront caractéristiques et très utilisés, tandis que d'autres ne le seront pas. Les lignes de la matrice d'utilisation des noyaux non-utilisés sont blanches, et nous pouvons les supprimer pour obtenir un dictionnaire plus compact. Par exemple, le noyau $l = 8$ de notre dictionnaire pourrait être supprimé (Fig. 4.4). Ainsi, il est préférable de légèrement surestimer L.
Au final, la question cruciale est le choix du paramètre K. En fait, ce choix est empirique, étant donné qu'il dépend du nombre de primitives que l'utilisateur prévoit dans chaque signal de la base de données étudiée. Dans notre expérience, nous choisissons $K = 5$, car nous avons constaté empiriquement que 2–3 primitives principales codent l'information majeure, et que les autres codent les variabilités.

L'optimisation non-convexe du 2DRI-OMP / M-OMP et le traitement aléatoire des signaux d'entraînement donnent des dictionnaires différents à partir des mêmes hyperparamètres. Cependant, la variance des résultats est faible, et soit nous obtenons parfois les mêmes dictionnaires, soit ils ont des qualités similaires (rRMSE, taille du dictionnaire, interprétabilité). Pour la suites des expériences, remarquons qu'un dictionnaire appris orienté, *oriented learned dictionary* (OLD) en anglais, est aussi obtenu avec le M-DLA. Les trajectoires associées sont représentées en Fig. 4.5.

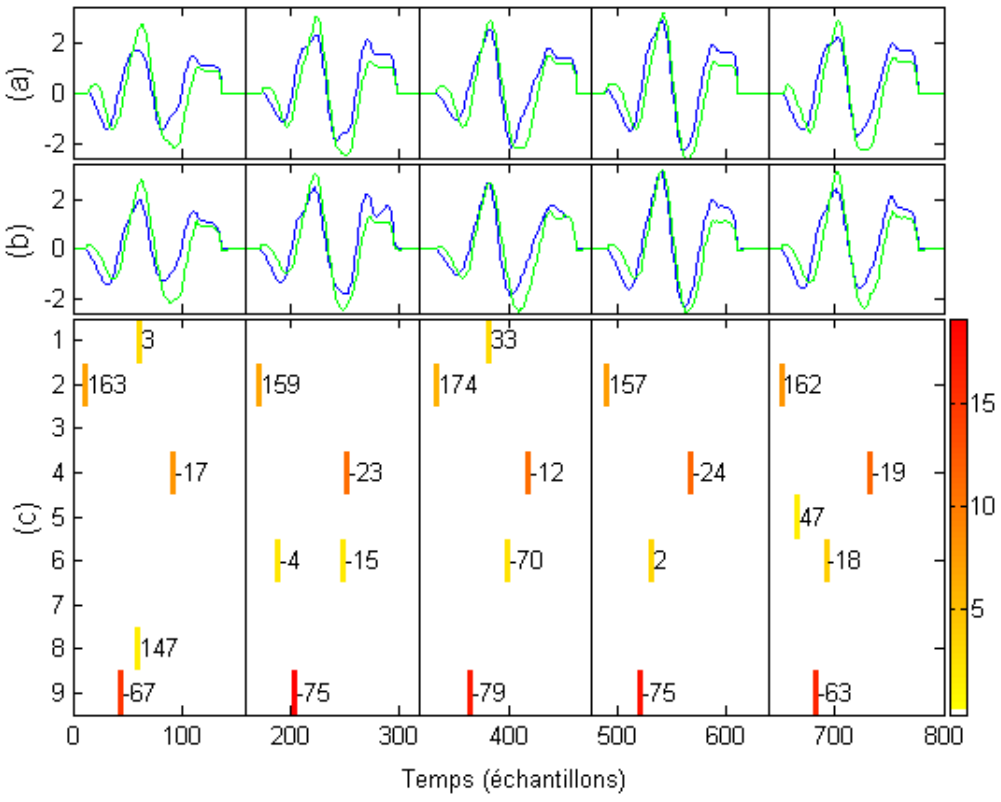


Fig. 4.6 – Signaux de vitesse originaux (a) et leurs approximations (b) pour cinq occurrences de la lettre *d*, et leur spikegramme associé (c).

4.3.3 Expérience 2 : décompositions des données

Pour évaluer la qualité de l'encodage parcimonieux, les décompositions non-orientées de cinq occurrences de la lettre d sur le NOLD sont étudiées en Fig. 4.6. Les signaux de vitesse originaux sont affichés en Fig. 4.6(a) et leurs approximations sont affichées en Fig. 4.6(b), avec les deux composantes $\mathfrak{v}_x$ et $\mathfrak{v}_y$. La rRMSE sur les vitesses est d'environ 12%, avec 4-5 atomes utilisés pour l'approximation. Les coefficients complexes $\left\{x_{l^k,\tau^k}\right\}_{k=1}^{K}$ sont affichés sur un spikegramme (Fig. 4.6(c)) condensant quatre informations :

- la position temporelle τ^k en abscisse,
- l'indice de noyaux l^k en ordonnée,
- l'amplitude du coefficient $\left|x_{l^k,\tau^k}\right|$ en échelle de couleur,
- l'angle de rotation θ_{l^k,τ^k} par la valeur réelle à côté, donné en degré.

Le faible nombre d'atomes utilisés pour l'approximation du signal montre la parcimonie de la décomposition, qui est appelé *sparse code* en anglais. Les atomes principaux sont ceux de plus fortes amplitudes, comme les noyaux $l = 2$, $l = 4$ et $l = 9$, et ils concentrent une grande part de l'énergie. Les atomes secondaires codent les variabilités entre les différentes réalisations de la même lettre. La reproductibilité de la décomposition est soulignée par la répétition des valeurs d'amplitude et d'angle pour les différentes occurrences. La parcimonie et la reproductibilité sont les preuves d'un dictionnaire adapté.

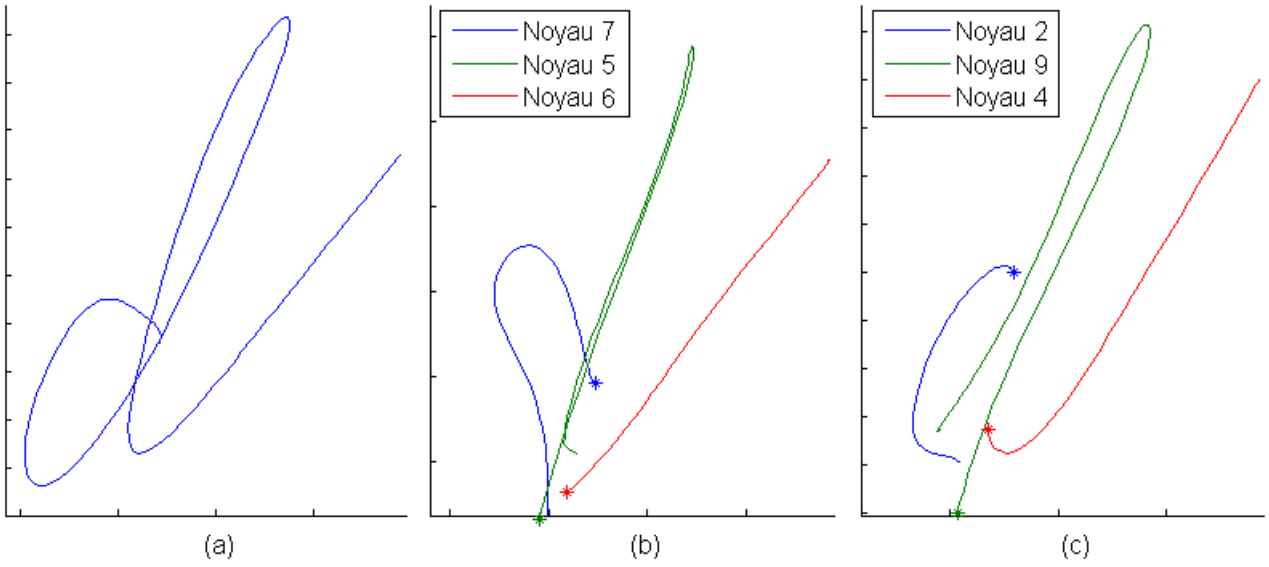


Fig. 4.7 – Lettre d : trajectoire originale (a), trajectoire reconstruite orientée (b) et trajectoire reconstruite non-orientée (c).

La trajectoire de la lettre d originale en Fig. 4.7(a) (*resp.* p en Fig. 4.8(a)) est maintenant reconstruite avec ses atomes principaux. Nous comparons le cas orienté en Fig. 4.7(b) (*resp.* Fig. 4.8(b)) en utilisant le OLD et le cas non-orienté en Fig. 4.7(c) (*resp.* Fig. 4.8(c)) en utilisant le NOLD. Comme exemple de reconstruction, la trajectoire de la lettre d de la Fig. 4.7(c) est reconstruite comme la somme des noyaux NOLD $l = 2$, $l = 4$ et $l = 9$ (les trajectoires sont en Fig. 4.3), qui sont spécifiés par les amplitudes et les angles du spikegramme de la Fig. 4.6(c). Maintenant, nous nous intéressons à la barre verticale qui est commune aux lettres d et p (Fig. 4.7(a) et Fig. 4.8(a)). Pour la coder, le cas orienté utilise deux noyaux différents : le noyau $l = 5$ pour d (Fig. 4.7(b), en vert) et le noyau $l = 12$ pour p (Fig. 4.8(b), en vert).

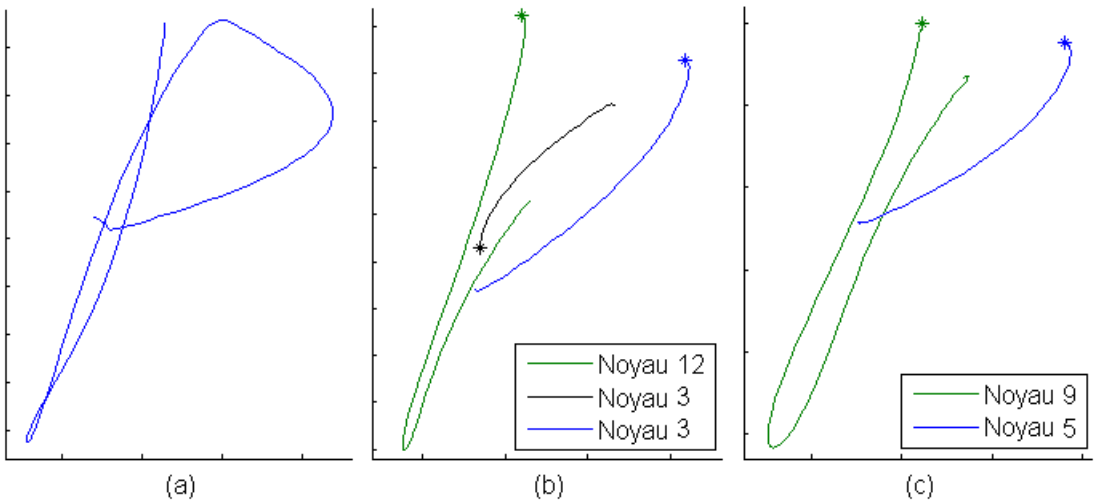


Fig. 4.8 – Lettre p : trajectoire originale (a), trajectoire reconstruite orientée (b) et trajectoire reconstruite non-orientée (c).

Au contraire, le cas non-orienté n'a besoin que d'un seul noyau pour ces deux lettres : le noyau $l = 9$ (Fig. 4.7(c) et Fig. 4.8(c), en vert), qui est utilisé avec une différence d'angles de rotation moyenne de 180°. Ainsi, l'approche non-orientée réduit la redondance du dictionnaire et fournit un dictionnaire de noyaux capables de tourner encore plus compact. La détection des invariants par rotation permet de faire décroître la taille du dictionnaire de $L = 12$ pour le OLD à $L = 9$ pour le NOLD.

4.3.4 Expérience 3 : décompositions des données tournées

Afin de simuler la rotation de la tablette d'acquisition, nous tournons artificiellement les données de l'ensemble de test. Les lettres sont désormais tournées d'un angle de -45° et de -90°.

Nous décomposons les signaux de test sur les deux dictionnaires (NOLD et OLD) appris en Section 4.3.2. La Fig. 4.9 montre les décompositions non-orientées des seconde et troisième occurrences des exemples utilisés en Fig. 4.6. Les signaux de vitesse tournés de -45° en Fig. 4.9(a) (*resp.* -90°, Fig. 4.9(d)) sont approximés dans le cas non-orienté en Fig. 4.9(b) (*resp.* Fig. 4.9(e)). Dans ces deux cas, la rRMSE est identique à l'expérience précédente, quand les lettres ne sont pas tournées. La Fig. 4.9(c) (*resp.* Fig. 4.9(f)) montre le spikegramme associé. Les différences d'angles des atomes principaux entre les spikegrammes des Fig. 4.6(c), Fig. 4.9(c) et Fig. 4.9(f) correspondent aux perturbations angulaires que nous avons appliquées, soit -45° et -90°. Cela montre l'invariance à la rotation de la décomposition 2DRI.

La trajectoire de la lettre d tournée de -90° en Fig. 4.10(a) est reconstruite avec ses atomes principaux, en comparant le cas orienté en Fig. 4.10(b) en utilisant le OLD, et le cas non-orienté en Fig. 4.10(c) en utilisant le NOLD. Dans le cas orienté, la rRMSE passe de 15% en Fig. 4.7(b) à 30% en Fig. 4.10(b), et l'encodage parcimonieux est moins efficace. De plus, les noyaux (*a fortiori* les atomes) sélectionnés sont différents : il n'y a plus de reproductibilité. La différence entre ces deux reconstructions montre la nécessité d'être robuste à la rotation. Dans le cas non-orienté, la rRMSE est identique quel que soit l'angle de rotation (Fig. 4.7(c) et Fig. 4.10(c)), et elle est toujours plus faible que dans le cas orienté. Les atomes sélectionnés sont identiques ce qui montre l'invariance par rotation de la décomposition.

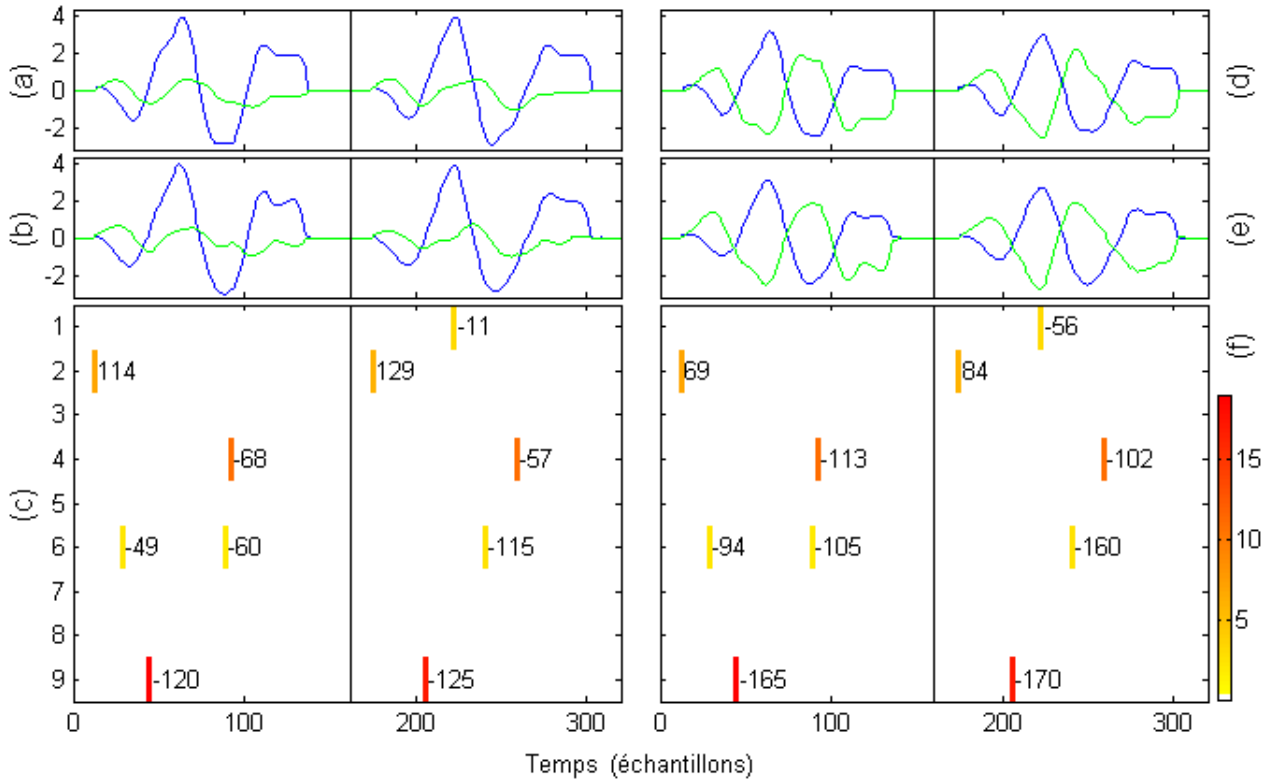


Fig. 4.9 – Signaux de vitesse tournés de -45° (a) (*resp.* -90° (d)) et leurs approximations (b) (*resp.* (e)) pour deux occurrences de la lettre *d*, et leurs spikegrammes (c) (*resp.* (f)).

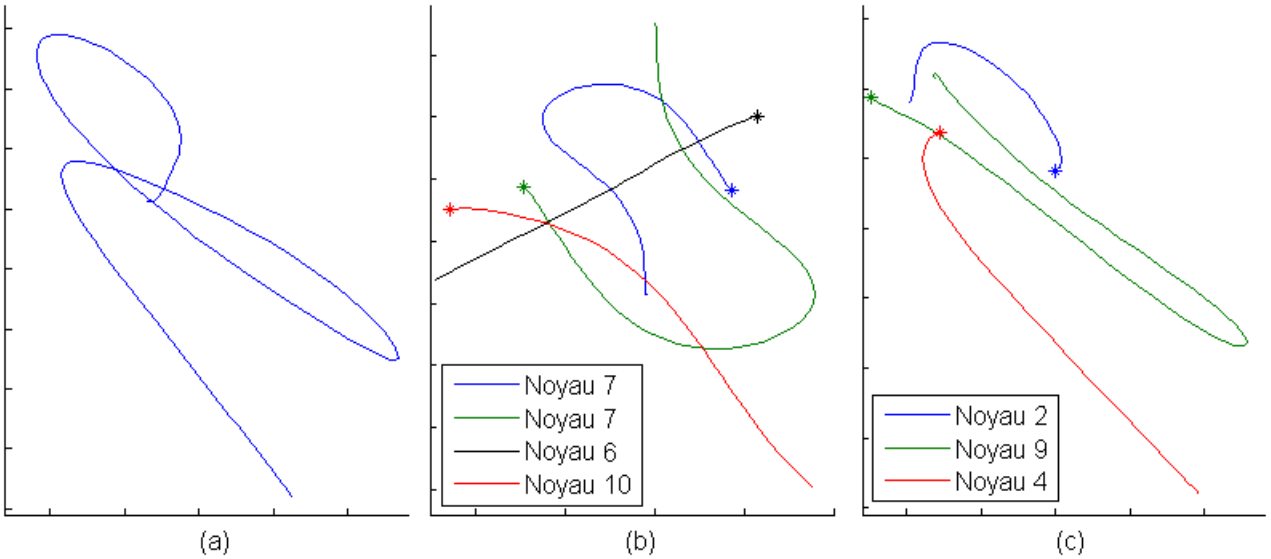


Fig. 4.10 – Lettre *d* tournée d'un angle -90° : trajectoire initiale (a), trajectoire reconstruite orientée (b) et trajectoire reconstruite non-orientée (c).

Pour conclure cette section, les méthodes ont été validées sur des signaux bivariés et ont montré l'apport de l'encodage parcimonieux invariant par translation et par rotation 2D.

4.4 Comparaisons

Deux comparaisons sont faites dans cette section : les dictionnaires appris sont d'abord comparés aux dictionnaires classiques, puis nous comparons les apprentissages orientés et non-orientés.

4.4.1 Comparaison avec des dictionnaires classiques

Dans cette section, l'ensemble de test est utilisé pour la comparaison, mais sans rotation. Seule la composante $\mathfrak{v}_x$ est considérée, pour être comparée aux autres méthodes dans un cas unicomposante. Nous comparons les dictionnaires appris précédemment pour les approches non-orientée (le NOLD, avec $L = 9$) et orientée (le OLD, avec $L = 12$) avec les dictionnaires paramétriques classiques issus de transformées : la transformée de Fourier discrète (*discrete Fourier transform* DFT), la transformée en cosinus discrète, (*discrete cosine transform* DCT), et la transformée en ondelettes biorthogonales [3] (*biorthogonal wavelet transform* BWT). Pour chaque dictionnaire, les approximations K-parcimonieuses sont calculées sur l'ensemble de test, et le taux de reconstruction ρ est calculé ainsi :

$$\rho = 1 - \frac{1}{Q} \sum_{q=1}^{Q} \frac{\|\epsilon_q\|_2}{\|y_q\|_2} \,. \tag{4.13}$$

Le taux ρ est représenté en fonction de la parcimonie K en Fig. 4.11.

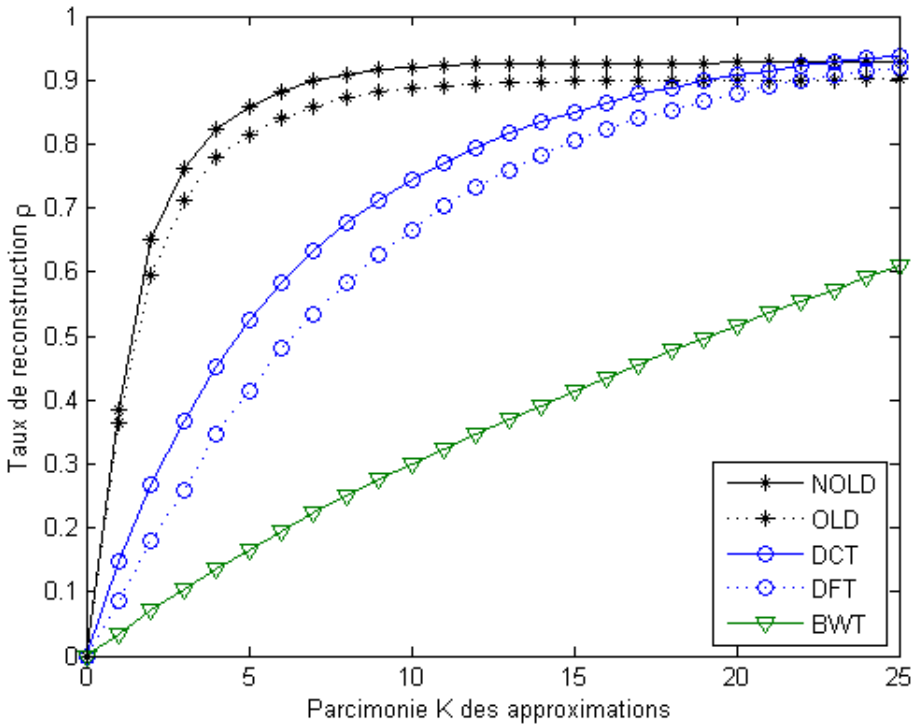


Fig. 4.11 – Taux de reconstruction ρ sur l'ensemble de test en fonction de la parcimonie K des approximations pour les différents dictionnaires.

3. comme les différents types d'ondelettes donnent des performances similaires, nous présentons seulement la CDF 9/7

Nous voyons que pour peu de coefficients, les signaux sont mieux reconstruits avec les dictionnaires appris (NOLD $L = 9$ et OLD $L = 12$) qu'avec ceux à bases de Fourier (DFT et DCT), eux-mêmes meilleurs que ceux à base d'ondelettes (BWT). Ces résultats montrent l'efficacité de représentation des dictionnaires appris par rapport aux dictionnaires paramétriques. Si l'apprentissage de dictionnaire est long comparé aux transformées rapides, il est calculé une seule fois pour une application donnée.

Pour le NOLD, seulement 7 atomes sont requis pour atteindre un taux de 90%, et l'asymptote est à 93%. De plus, ρ_{NOLD} est toujours supérieur à ρ_{OLD} quelle que soit la valeur de K. L'invariance par rotation est donc utile même sans données tournées : elle fournit une meilleure adaptation aux variabilités entre les différentes réalisations. Les taux au-delà de $K = 25$ ne sont pas représentés en Fig. 4.11. Si c'était le cas, nous pourrions voir qu'ils atteignent un taux de reconstruction de 100% : ils génèrent tout l'espace contrairement aux dictionnaires appris. Les dictionnaires génériques sont des *bases de l'espace* alors que les dictionnaires appris peuvent être considérés comme des *bases de l'énergie*. Dans les DLAs, l'approximation parcimonieuse sélectionne les motifs de forte énergie, et ces motifs sont ensuite appris. Ainsi, les parties du signal jamais sélectionnées ne sont jamais apprises.

4.4.2 Comparaison entre les apprentissages orientés et non-orientés

Dans la Section 4.3.4, nous évaluons seulement l'invariance par rotation des décompositions avec des données tournées, mais pas l'invariance par rotation de l'apprentissage. Les données de l'ensemble de test ont été tournées, mais pas celle de l'ensemble d'apprentissage. Ici, nous proposons d'étudier l'invariance par rotation de la méthode d'apprentissage avec des signaux d'apprentissage tournés.

Dans cette comparaison, l'apprentissage et les décompositions sont effectués sur les ensembles de données $\boldsymbol{Y}$ (contenant l'ensemble d'apprentissage et l'ensemble de test), qui sont tournés à différents angles. $\boldsymbol{Y}_1$ contient les données originales, $\boldsymbol{Y}_2$ contient les données originales et les données tournées de 120°, $\boldsymbol{Y}_3$ contient les données originales et tournées de 120° et 240°, et $\boldsymbol{Y}_4$ contient les données originales et les données tournées d'angles aléatoires. Les ensembles d'apprentissage permettent d'apprendre différents dictionnaires : le NOLD avec $L = 9$ noyaux et le OLD avec $L = 12$, 18, 24 et 30 noyaux. Les décompositions sur les ensembles de test donnent le taux de reconstruction ρ pour $K = 5$.

TABLE 4.1 – Résultats des taux de reconstruction des signaux de vitesse de l'ensemble de test.

ρ (%)	$\boldsymbol{Y}_1$	$\boldsymbol{Y}_2$	$\boldsymbol{Y}_3$	$\boldsymbol{Y}_4$
NOLD $L=9$	85.8	85.6	85.9	85.0
OLD $L=12$	81.6	79.6	77.0	77.5
OLD $L=18$	83.0	81.4	79.98	78.9
OLD $L=24$	83.9	82.6	81.3	79.4
OLD $L=30$	84.8	83.5	82.8	80.7

La Table 4.1 donne les résultats des taux de reconstruction en fonction des ensembles de données (colonnes) et du type de dictionnaire (lignes). Pour l'apprentissage non-orienté, les résultats sont similaires pour tous les ensembles de données. Pour les apprentissages orientés, la qualité des approximations augmente avec le nombre de noyaux L : les noyaux supplémentaires permettent de mieux générer l'es-

pace. Cependant, même avec 30 noyaux, le OLD montre des résultats moins bons que le NOLD avec seulement 9 noyaux. De plus, le taux de reconstruction décroît quand le nombre d'angles différents dans les données augmente, puisque les lettres tournées sont considérées comme de nouvelles lettres.

Ces résultats permettent seulement de voir la qualité des approximations, mais pas l'invariance par rotation ni la reproductibilité des décompositions. Pour ce faire, un critère de similarité est mis en place, en utilisant la matrice d'utilisation. Comme expliqué en Section 4.3.2, cette matrice est formée en calculant les moyennes des valeurs absolues des coefficients des décompositions.

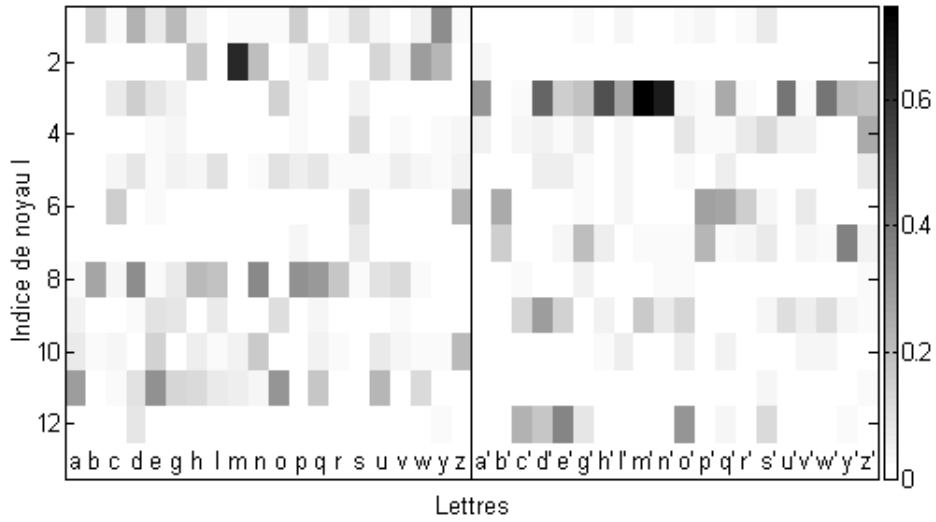


Fig. 4.12 – Matrice d'utilisation pour le OLD ($L = 12$) sur l'ensemble $\boldsymbol{Y}_2$. La moyenne des valeurs absolues des coefficients est donnée en fonction de l'indice de noyau l et de la lettre du signal. Les lettres avec $'$ sont celles qui sont tournées.

En Fig. 4.12, les valeurs sont données en fonction de l'indice de noyau l (ordonnée) et de la lettre du signal (abscisse). La Fig. 4.12 montre la matrice d'utilisation calculée sur $\boldsymbol{Y}_2$ pour le OLD, avec $L = 12$. Nous voyons que ce ne sont pas les mêmes noyaux qui sont utilisés pour coder une lettre et sa version tournée, notée $(.)'$, comme la lettre d sur les Fig. 4.7(b) et 4.10(b). Pour évaluer ce phénomène, le critère de similarité c est défini comme la moyenne des produits scalaires normalisés entre la colonne d'une lettre et celle de sa version tournée.

TABLE 4.2 – Résultats des critères de similarité des matrices d'utilisation de l'ensemble de test.

c (%)	$\boldsymbol{Y}_2$	$\boldsymbol{Y}_3$	$\boldsymbol{Y}_4$
NOLD $L=9$	100	100	100
OLD $L=12$	18.7	24.7	67.3
OLD $L=18$	14.1	17.2	60.6
OLD $L=24$	15.2	12.3	58.8
OLD $L=30$	6.3	9.0	57.8

La Table 4.2 résume le produit scalaire moyen c donné en pourcentage en fonction des ensembles

de données (colonnes) et du type de dictionnaire (lignes). La définition du critère et la composition des ensembles de test ont été choisies pour donner $c = 100\%$ dans le cas de référence non-orienté. Il reste à 100% quel que soit l'ensemble de données, ce qui montre l'invariance par rotation. Cependant, en réalité, il n'est pas utile de réaliser l'apprentissage sur des données tournées. Comme constaté en Section 4.3.4, l'apprentissage non-orienté sur les données originales est suffisant pour fournir un dictionnaire adapté qui est robuste aux rotations.

Pour les apprentissages orientés, si les dictionnaires de plus grandes tailles donnent les meilleurs taux de reconstruction (Table 4.1), ils ont les critères de similarité les plus mauvais, puisque un trop grand nombre de noyaux a tendance à disperser l'information. Ainsi, augmenter artificiellement la taille du dictionnaire n'est pas une bonne idée pour l'encodage parcimonieux, parce que cela dégrade les résultats en terme de reproductibilité des décompositions. De plus, augmenter le nombre d'angles différents dans les données donne de meilleures reproductibilités, puisque les signaux n'influencent plus l'apprentissage par une orientation fixe et, par conséquent, les noyaux orientés sont plus généraux.

4.5 Discussions

4.5.1 Considérations sur le modèle 2DRI

L'apprentissage de dictionnaire permet d'extraire les primitives des signaux. Le dictionnaire résultant peut être vu comme un catalogue de motifs élémentaires qui sont dédiés à l'application considérée et qui ont un sens physique, contrairement aux dictionnaires classiques comme les ondelettes, curvelettes, etc. Ainsi, les décompositions faites sur un tel dictionnaire sont la combinaison parcimonieuse de causes sous-jacentes. L'erreur faible des décompositions parcimonieuses montre l'efficacité de l'encodage parcimonieux.

L'approche non-orientée réduit la taille du dictionnaire L de deux manières :

- quand les signaux ne peuvent pas tourner, l'approche non-orientée détecte les invariants par rotation (la barre verticale des lettres d et p par exemple), ce qui réduit la taille du dictionnaire.
- quand les signaux peuvent tourner, pour fournir un encodage parcimonieux efficace, l'approche orientée doit apprendre les primitives du mouvement pour tous les angles possibles. A l'inverse, un seul apprentissage est suffisant pour l'approche non-orientée : cela produit une vraie diminution de la taille du dictionnaire.

De cette manière, le cas d'invariance par translation et par rotation fournit un dictionnaire appris $\mathbf{\Psi}$ compact. De plus, l'approche non-orientée permet la robustesse à n'importe quelle direction d'écriture (rotation de la tablette d'acquisition) et n'importe quelle inclinaison d'écriture (variabilités intra et inter utilisateurs). En ajoutant une étape de traitement, l'information sur les angles permet de déterminer l'angle de la direction d'écriture.

En résumé, avec ce modèle de représentation invariante par translation temporelle et par rotation 2D, chaque noyau est potentiellement démultiplié en une famille d'atomes translatés à tous les échantillons et tournés à tous les angles. Ainsi, le dictionnaire de noyaux invariants génère un dictionnaire d'atomes très redondant, qui est donc idéal pour représenter les données étudiées redondantes.

4.5.2 Limitations de la méthode

En regardant les décompositions non-orientées d'autres lettres, nous observons des problèmes de stabilité. Nous étudions plus particulièrement deux occurrences de la lettre v. Nous affichons la trajectoire originale de la première (*resp.* deuxième) occurrence de la lettre v en Fig. 4.13(a) (*resp.* Fig. 4.14(a)), et sa trajectoire reconstruite non-orientée en Fig. 4.13(b) (*resp.* Fig. 4.14(b)). Nous observons que les deux occurrences n'ont pas les mêmes décompositions : elles n'utilisent pas les mêmes noyaux aux mêmes moments. La décomposition de la première occurrence est optimale, alors que celle de la deuxième est un minimum local. Pour la deuxième occurrence, à la première itération, le 2DRI-OMP a sélectionné le noyau $l = 7$ pour représenter au mieux la lettre v, dont la forme est plus serrée que la première occurrence. Cette différence a pour origine la non-convexité de l'OMP qui est une minimisation ℓ_0. Cet algorithme est dit glouton car il espère obtenir l'optimum global par des choix successifs d'optima locaux. L'inconvénient majeur de ce phénomène est la non-reproductibilité des décompositions.

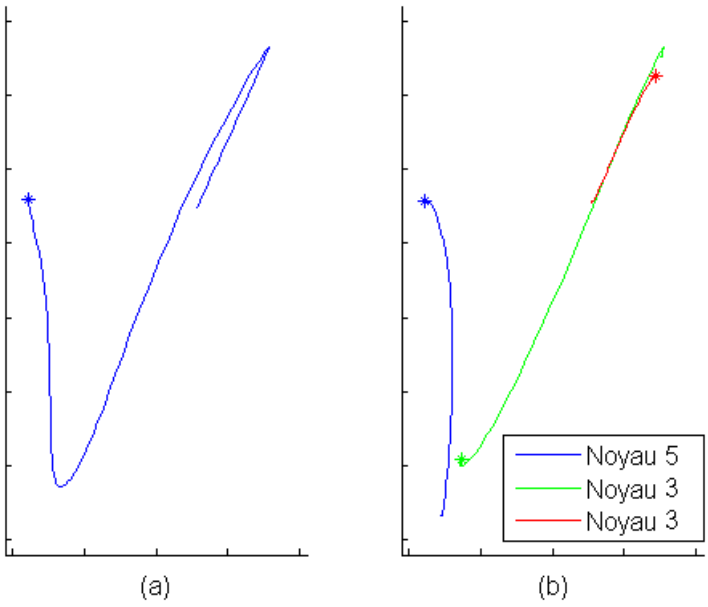


Fig. 4.13 – Première occurrence de la lettre v : trajectoire originale (a) et trajectoire reconstruite non-orientée (b).

Comme expliqué en Section 1.3.2 à propos des méthodes d'apprentissage *online*, dans un processus itératif, chaque étape n'a pas besoin d'être minimisée parfaitement pour atteindre l'optimum global. Ainsi, incluse dans un DLA, une poursuite ℓ_0 ne pénalise pas l'apprentissage du dictionnaire. Mais une fois le dictionnaire appris, les décompositions ultérieures réalisées avec le dictionnaire appris et avec une poursuite ℓ_0 peuvent s'avérer être des solutions locales, comme constaté ici avec la lettre v. L'idéal est donc d'utiliser une méthode de type ℓ_0 durant l'apprentissage, qui est rapide et dont la non-convexité n'est pas pénalisante, puis d'utiliser une méthode de type ℓ_1 pour effectuer les décompositions adaptées sur le dictionnaire appris.

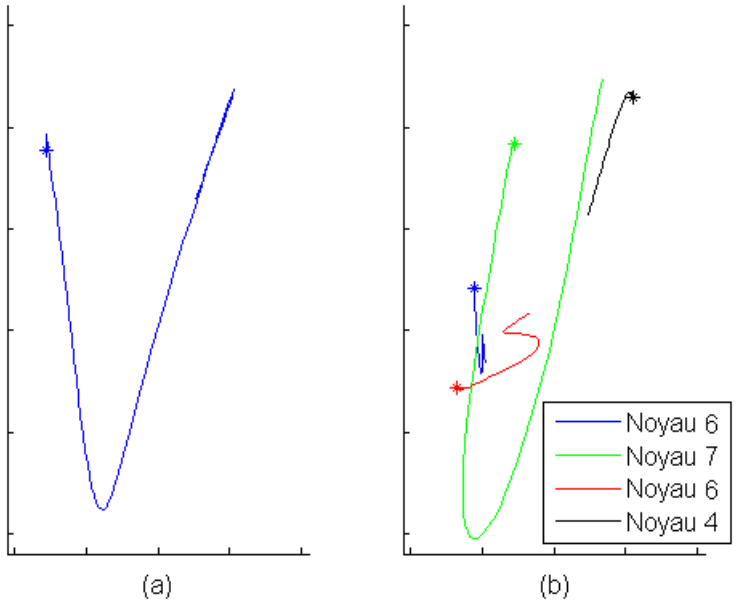


Fig. 4.14 – Deuxième occurrence de la lettre v : trajectoire originale (a) et trajectoire reconstruite non-orientée (b).

Conclusion

Considérant des signaux bivariés, nous voulons que leurs représentations soient indépendantes de l'orientation du mouvement dans l'espace 2D. Pour fournir une décomposition invariante par rotation 2D, les méthodes multivariées sont spécifiées au cas d'invariance par rotation 2D sous les noms de 2DRI-OMP et 2DRI-DLA. L'invariance par rotation est utile, mais pas seulement quand les données sont tournées, étant donné qu'elle permet de mieux coder les variabilités. Ces méthodes sont appliquées avec succès à des données d'écriture manuscrite.

Bibliographie

[Bra83] D.H. Brandwood : A complex gradient operator and its application in adaptive array theory. *IEE Proceedings F - Communications, Radar and Signal Processing*, 130:11–16, 1983.

[DL12] Z. Dai et J. Lücke : Unsupervised learning of translation invariant occlusive components. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR*, pages 2400–2407, 2012.

[FA10] A. Frank et A. Asuncion : UCI machine learning repository, 2010.

[KSU10] T. Kim, G. Shakhnarovich et R. Urtasun : Sparse coding for learning interpretable spatio-temporal primitives. *In Advances in Neural Information Processing Systems NIPS*, pages 1117–1125, 2010.

[MS11] M. Mørup et M. Schmidt : Transformation invariant sparse coding. *In Proc. Machine Learning for Signal Processing MLSP '11*, pages 1–6, 2011.

[MTS12] F. Meier, E. Theodorou et S. Schaal : Movement segmentation and recognition for imitation learning. *In JMLR W&CP*, volume 22, pages 761–769, 2012.

[MTSS11] F. Meier, E. Theodorou, F. Stulp et S. Schaal : Movement segmentation using a primitive library. *In IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2011*, pages 3407–3412, 2011.

[SS85] J.T. Schwartz et M. Sharir : Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. Rapport technique Robotics Report 46, Courant Institute of Mathematical Sciences, New York University, 1985.

[SSAS11] S. Satpathy, L. Sharma, A.K. Akasapu et N. Sharma : Towards mining approaches for trajectory data. *Int. J. of Advances in Science and Technology*, 2:38–43, 2011.

[VEG12] C. Vollmer, J.P. Eggert et H.-M. Gross : Modeling human motion trajectories by sparse activation of motion primitives learned from unpartitioned data. *In KI 2012 : Advances in Artificial Intelligence*, volume 7526 de *Lecture Notes in Computer Science*, pages 168–179, 2012.

[VGD04] M. Vlachos, D. Gunopulos et G. Das : Rotation invariant distance measures for trajectories. *In Proc. SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 707–712, 2004.

[WEK03] H. Wersing, J. Eggert et E. Körner : Sparse coding with invariance constraints. *In Proc. Int. Conf. Artificial Neural Networks ICANN*, pages 385–392, 2003.

[WTS07] B.H. Williams, M. Toussaint et A.J. Storkey : A primitive based generative model to infer timing information in unpartitioned handwriting data. *In Proc. Int. Joint Conf. on Artificial Intelligence IJCAI*, pages 1119–1124, 2007.

[WTS08] B.H. Williams, M. Toussaint et A.J. Storkey : Modelling motion primitives and their timing in biologically executed movements. *In Advances in Neural Information Processing Systems NIPS*, volume 20, pages 1609–1616, 2008.

Chapitre 5

Représentations invariantes par rotation 3D

Introduction

Dans le chapitre précédent, nous avons étudié l'invariance par rotation 2D qui est adaptée pour l'analyse de signaux de mouvements plans. Il est naturel de se poser la question concernant les mouvements en général, *i.e.* effectués sans restriction dans l'espace 3D. C'est pourquoi, dans ce chapitre, nous étudions des signaux trivariés pour les mouvements 3D. Nous souhaitons mettre en place une décomposition invariante par rotation 3D (en anglais, *3D rotation invariant* (3DRI)) afin que la représentation de ces signaux trivariés soit indépendante de leurs orientations. Ce chapitre peut donc être vu comme l'extension au cas 3D du chapitre précédent, afin de satisfaire les mêmes objectifs que le cas 2D. Cependant, cette extension n'est pas triviale : elle va poser de nouveaux verrous et nous montrerons lesquels et comment les lever.

Après un état de l'art sur les décompositions 3D effectué à la frontière de plusieurs domaines, nous présenterons les méthodes de décomposition et d'apprentissage invariants par rotation 3D. Ces méthodes seront comparées et validées sur des données simulées, puis illustrées sur des données réelles de Langage Parlé Complété.

5.1 Invariance par rotation 3D

Nous traitons désormais des données réelles trivariées, *i.e.* avec $V = 3$. Toutes les variables sont transposées par rapport au précédent chapitre. Nous étudions un signal trivarié $\boldsymbol{y} \in \mathbb{R}^{3\times N}$ composé de N échantillons, et un dictionnaire normé $\boldsymbol{\Psi}$ de noyaux multivariés $\boldsymbol{\psi}_l \in \mathbb{R}^{3\times T_l}$. Nous définissons le produit scalaire comme $\langle A, B\rangle = \mathrm{Tr}(AB^T)$ et la norme de Frobenius associée est notée $\|.\|_F$.

Pour obtenir l'invariance par rotation 3D, nous introduisons une matrice de rotation $R_{l,\tau} \in \mathbb{R}^{3\times 3}$ pour chaque atome trivarié $\boldsymbol{\psi}_l(t-\tau)$:

$$\boldsymbol{y}(t) = \sum_{l=1}^{L} \sum_{\tau\in\sigma_l} x_{l,\tau}\, R_{l,\tau}\, \boldsymbol{\psi}_l(t-\tau) + \boldsymbol{\epsilon}(t)\,. \tag{5.1}$$

Désormais, il faut estimer les coefficients x et les matrices de rotation $\mathcal{R}$ sous contrainte d'orthogonalité.

L'approximation parcimonieuse multivariée (2.12) spécifiée au cas de l'invariance par rotation 3D devient :

$$\min_{x,\mathcal{R}} \left\| \boldsymbol{y}(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau}\, R_{l,\tau}\, \boldsymbol{\psi}_l(t-\tau) \right\|_F^2$$

$$\text{t.q. } \|x\|_0 \leq K \quad \text{et} \quad \forall l \in \mathbb{N}_L, \forall \tau \in \sigma_l,\ R_{l,\tau} R_{l,\tau}^T = Id\,. \tag{5.2}$$

Nous rappelons ici que nous travaillons avec des données tricomposantes et non tridimensionnelles. Un signal temporel $\boldsymbol{y} \in \mathbb{R}^{3\times N}$ avec 3 composantes est abusivement qualifié de tridimensionnel. En fait, ce signal est tricomposant (tricanal ou trivarié selon le modèle d'analyse choisi) et non tridimensionel. Un signal tridimensionnel (dimension 3) serait $\boldsymbol{y} \in \mathbb{R}^{N_1\times N_2\times N_3}$ tel que les images vidéo, les images hyperspectrales ou d'autres données cubiques.

5.2 Etat de l'art des décompositions 3D

Différentes communautés (vision par ordinateur, traitement du signal, statistique, robotique et apprentissage) sont intéressées par la rotation 3D de motifs. Avec des terminologies différentes, ces domaines traitent souvent des mêmes problèmes. Dans un premier temps, nous faisons une revue des modèles de décomposition de trajectoires 3D, et ensuite nous étudierons les problèmes proches du problème d'approximation parcimonieuse (5.2).

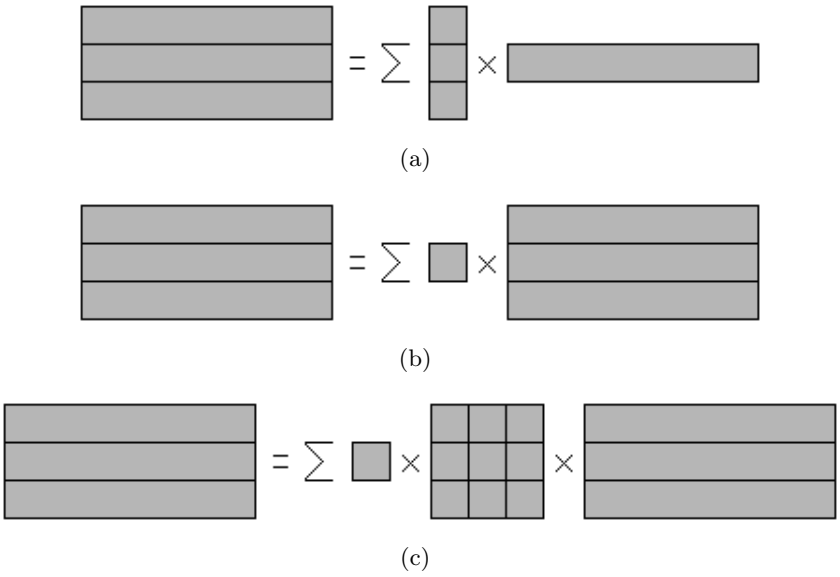
5.2.1 Modèles de décompositions tricomposantes

Dans un espace 3D, un signal tricomposantes dessine une trajectoire 3D composée de N échantillons temporels. Elle est décomposée sur des motifs élémentaires, et est donc décrite comme la somme de K vecteurs de bases. Plusieurs modèles peuvent être considérés pour étudier cette trajectoire.

En vision par ordinateur, Akhter *et al.* décrivent un objet 3D non rigide composé de P points comme P trajectoires temporelles composées de N échantillons [ASKK11]. Dans un premier temps, nous considérerons la trajectoire 3D d'un seul point. La trajectoire 3D $\boldsymbol{y} \in \mathbb{R}^{3\times N}$ est définie telle que :

$$\boldsymbol{y} = \sum_{k=1}^{K} a_k\, f_k\,, \tag{5.3}$$

avec $a_k \in \mathbb{R}^{3\times 1}$ les coefficients, et $f_k \in \mathbb{R}^{1\times N}$ les vecteurs de trajectoire. Ainsi, la trajectoire $\boldsymbol{y}$ est la somme de K vecteurs de trajectoire $[f_k]_{k=1}^K$. La dualité entre le modèle fondé sur une base de vecteurs de trajectoire et celui fondé sur une base de vecteurs de forme [BHB00] est démontrée dans [ASKK11]. Le modèle (5.3) s'avère être un modèle multicanal, comme évoqué au Chapitre 2. Akhter *et al.* utilisent des bases telles que la DCT ou issus de la PCA dans leurs décompositions [ASKK11], et ils veulent que ces décompositions soient compactes. Cependant, ils ignorent comment choisir la constante K et comment sélectionner K vecteurs parmi les N possibles. Ainsi, un compromis manuel (entre K et l'erreur résiduelle) est utilisé sur une recherche exhaustive parmi toutes les possibilités pour déterminer les vecteurs optimaux. Pour résoudre ce problème, une approximation parcimonieuse multicanale sera utilisée pour nos expériences comparatives.



(a)

(b)

(c)

Fig. 5.1 – Illustration des trois modèles pour décrire une trajectoire 3D : le modèle multicanal (a) décrit en Eq. (5.3), le modèle multivarié (b) décrit en Eq. (5.4) et le modèle 3DRI (c) décrit en Eq. (5.5).

En traitement du signal, un signal temporel multicomposante est décrit dans le Chapitre 2 comme la somme d'atomes multivariés. En considérant ici le cas particulier tricomposantes, la trajectoire 3D de N échantillons est vue comme la somme de K trajectoires 3D. La trajectoire $\boldsymbol{y} \in \mathbb{R}^{3\times N}$ est définie comme :

$$\boldsymbol{y} = \sum_{k=1}^{K} x_k \, \boldsymbol{\phi}_k \, , \tag{5.4}$$

avec $x_k \in \mathbb{R}$ les coefficients, et $\boldsymbol{\phi}_k \in \mathbb{R}^{3\times N}$ les atomes 3D. Comme expliqué en Chapitre 2, ce modèle multivarié multiplie chaque trajectoire trivariée $\boldsymbol{\phi}_k$ par un coefficient alors que le modèle multicanal multiplie chaque trajectoire monocomposante f_k par trois coefficients. Le modèle trivarié 5.4 est résolu en utilisant le M-OMP introduit au Chapitre 2 qui sélectionne K atomes actifs parmi les M atomes possibles du dictionnaire.

Le but de ce chapitre est de fournir un modèle incluant une invariance à la rotation 3D, et il sera nommé 3DRI. Ainsi, une matrice de rotation $R_k \in \mathbb{R}^{3\times 3}$ est appliquée à chaque atome $\boldsymbol{\phi}_k$ et le modèle (5.4) devient :

$$\boldsymbol{y} = \sum_{k=1}^{K} x_k \, R_k \, \boldsymbol{\phi}_k \, . \tag{5.5}$$

Chaque matrice de rotation R_k doit être orthogonale, *i.e.* elle doit vérifier la condition : $R_k R_k^T = Id$. La trajectoire $\boldsymbol{y}$ est représentée comme la somme de trajectoires 3D capables de tourner. Les différences entre ces trois modèles de décomposition de trajectoires 3D sont illustrées en Fig. 5.1.

Le modèle 3DRI (5.5) permet aux atomes de tourner, mais nous avons besoin d'une méthode d'approximation spécifique qui estiment aussi les matrices de rotation. Nous voulons donc estimer les coefficients $x = [x_k]_{k=1}^K$ et les matrices de rotation $\mathcal{R} = \{R_k\}_{k=1}^K$ au sens des moindres carrés. Le problème

s'écrit :

$$\min_{x,\mathcal{R}} \left\| \boldsymbol{y} - \sum_{k=1}^{K} x_k \, R_k \, \boldsymbol{\phi}_k \right\|_F^2 \text{ t.q. } \forall k \in \mathbb{N}_K, R_k R_k^T = Id \,. \tag{5.6}$$

Cette équation est une version simplifiée de l'Eq. (5.2) pour pouvoir se comparer à l'état de l'art.

5.2.2 Problèmes de minimisation 3D

Dans ce paragraphe, nous faisons la revue des problèmes de minimisation proches de notre problème d'intérêt (5.2).

Recalage 3D rigide ou problème de Procrustes orthogonal

Nous considérons ici une transformation rigide composée d'une rotation 3D rotation R et d'une translation spatiale T entre un atome trivarié $\boldsymbol{\phi}$ et le signal original $\boldsymbol{y}$. Le problème de Procrustes orthogonal consiste à trouver les paramètres R et T tel que :

$$\min_{R,T} \| \boldsymbol{y} - R \, \boldsymbol{\phi} - T \|_F^2 \text{ t.q. } RR^T = Id \,. \tag{5.7}$$

Eggert *et al.* [ELF97] font la revue des principales méthodes qui donnent une solution analytique à ce problème de recalage 3D rigide : par SVD [Sch66, Kan94], par les quaternions unitaires [Hor87], par matrice orthonormale [Hor88], et par les quaternions duaux [WSV91].

Le problème multivues reconstruit le signal $\boldsymbol{y}$ à partir de plusieurs observations $\boldsymbol{\phi}_k$ prises sous différents angles [KPJR91, BSGL96, BS98]. Comme les différentes prises de vues se recouvrent, le problème s'écrit :

$$\min_{\mathcal{R},\mathcal{T}} \left\| \boldsymbol{y} - \cup_{k=1}^{K} R_k \, \boldsymbol{\phi}_k + T_k \right\|_F^2 \text{ t.q. } \forall k \in \mathbb{N}_K, R_k R_k^T = Id \,, \tag{5.8}$$

avec R_k la matrice de rotation et T_k la translation associées à l'atome trivarié $\boldsymbol{\phi}_k$. Cette formulation est différente de notre problème (5.6) ; dans l'Eq. (5.8), l'union des atomes est considérée et non la somme comme dans l'Eq. (5.6). La différence entre ces deux problèmes est illustrée en Fig. 5.2 avec des motifs 1D (au lieu de 3D, pour une meilleure visualisation).

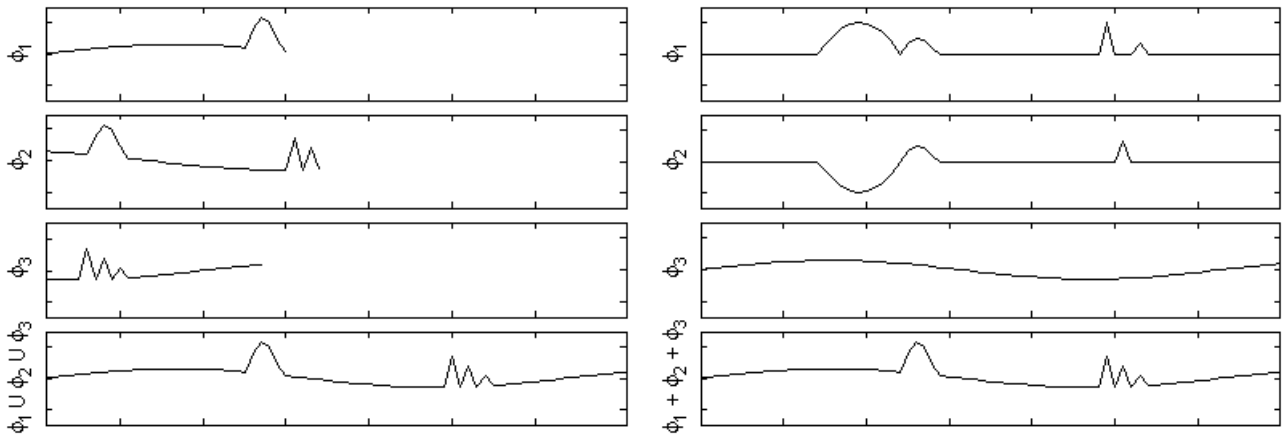
Problèmes de Procrustes généralisés

Considérant plusieurs éléments $\boldsymbol{\phi}_k$, l'analyse de Procrustes généralisée [Gow75, GD04, Gow10] résout le problème :

$$\min_{\mathcal{R}} \sum_{i<j}^{K} \left\| R_i \, \boldsymbol{\phi}_i - R_j \, \boldsymbol{\phi}_j \right\|_F^2 \text{ t.q. } \forall k \in \mathbb{N}_K, R_k R_k^T = Id \,, \tag{5.9}$$

avec $K \geq 2$. Ce problème, résolu en ajoutant des contraintes et en utilisant une moyenne groupée, est différent du problème (5.6).

Dans [GD04], Gower et Dijksterhuis font la revue de multiples problèmes de Procrustes différents et de beaucoup de généralisations. Cependant, ils ne parlent pas du problème (5.6).



(a) Union. (b) Somme.

Fig. 5.2 – Illustration de la différence entre l'union (a) et la somme (b) de trois motifs 1D $\{\phi_1, \phi_2, \phi_3\}$.

3D *matching*

Comparé au problème (5.7), la translation spatiale n'est plus considérée ici, et $\boldsymbol{\psi}(t)$ est un noyau translatable (complété avec des zéro pour avoir N échantillons). Le problème de robotique *3D curve matching* est formalisé par :

$$\min_{R,\tau} \| \boldsymbol{y}(t) - R\,\boldsymbol{\psi}(t-\tau) \|_F^2 \text{ t.q. } RR^T = Id\,, \tag{5.10}$$

avec τ le décalage d'échantillons. Ce problème est résolu en calculant la matrice de rotation optimale R (par une méthode basée sur une matrice orthonormale) pour chaque échantillon τ [SS85, BSSS86]. De plus, paramétrée de cette manière, la méthode n'est pas invariante à l'échelle puisqu'il n'y a pas de facteur d'échelle dans l'optimisation. Sans décalage τ, nous retrouvons simplement le problème de recalage 3D rigide (5.7).

Pour finir, nous évoquons l'algorithme *iterative closest point* (ICP) [BM92, RL01] qui permet de faire coïncider deux ensembles de points 3D, avec l'un étant un sous-ensemble de l'autre. Considérant $\boldsymbol{y}$ comme un ensemble de N points 3D, et $\boldsymbol{\phi}$ comme un ensemble de η points 3D, avec $\eta > N$, $\boldsymbol{\phi}_\Gamma$ est un sous-ensemble de $\boldsymbol{\phi}$ composé de N points 3D indexés par l'ensemble d'indices Γ. L'algorithme ICP résout le problème suivant :

$$\min_{\Gamma,R,T} \| \boldsymbol{y} - R\,\boldsymbol{\phi}_\Gamma - T \|_F^2 \text{ t.q. } RR^T = Id\,. \tag{5.11}$$

Cet algorithme utilise des signatures de forme pour identifier les N indices de Γ parmi les η possibles, et cette optimisation est fortement non-convexe.

Remarquons que tous les problèmes que nous venons d'évoquer dans cet état de l'art sont différents de notre problème (5.6). Si certaines optimisations évoquées précédemment s'appliquent autant à des formes qu'à des trajectoires, par la suite nous travaillerons sur des signaux de trajectoires.

5.2.3 Problème complet

En combinant maintenant les problèmes d'invariance par translation (cf. Section 1.1.3) et par rotation 3D, nous retrouvons l'Eq. (5.2) qui combine l'Eq. (2.12) dont l'optimisation est NP-difficile et l'Eq. (5.6)

dont nous ignorons s'il existe une solution analytique.
L'Eq. (5.2) a deux cas particuliers déjà résolus : quand $K = 1$, nous retrouvons le *3D curve matching* de l'Eq. (5.10) ; et quand $R_k = Id$, nous retrouvons l'approximation parcimonieuse de l'Eq. (2.12) avec des signaux trivariés, et elle est résolue par le M-OMP. Dans la suite de ce chapitre, nous proposons deux optimisations non-convexes pour résoudre ce problème particulièrement difficile.

La rotation 3D est un problème difficile, qui est souvent contourné au lieu d'être résolu. Pour faire de la reconnaissance de trajectoires, la plupart des méthodes n'estiment pas les paramètres de rotation, mais utilisent des descripteurs invariants à la rotation [CAS05, BMB+09], aussi appelés signatures de forme. Comme les paramètres de rotation ne sont pas calculés, cela engendre une perte d'information contrairement à la méthode que nous proposons ici.

5.3 MP invariant par rotation 3D

Dans cette section, nous détaillons d'abord la méthode choisie pour le recalage 3D qui sera l'étape centrale des algorithmes introduits. Ensuite, une optimisation non-convexe basée sur le MP est présentée pour résoudre le problème d'approximation parcimonieuse (5.2) et elle est appelée 3DRI-MP.

5.3.1 Recalage 3D par SVD

Le problème de recalage (5.7) est considéré ici avec un atome trivarié normé $\boldsymbol{\phi} \in \mathbb{R}^{3\times N}$ et un facteur d'échelle, mais sans translation spatiale. Les paramètres cherchés sont la rotation R et le facteur d'échelle x :

$$\min_{x,R} \| \boldsymbol{y} - x\, R\, \boldsymbol{\phi} \|_F^2 \text{ t.q. } RR^T = Id\,. \tag{5.12}$$

Pour résoudre ce problème de recalage 3D, la méthode par SVD est choisie parmi les autres parce que c'est la moins coûteuse d'un point de vue calculatoire, et parce qu'elle traite facilement les cas particuliers du bruit et de motifs coplanaires [ELF97]. Introduite par Schonemann [Sch66] pour résoudre le problème de Procrustes orthogonal, la méthode par SVD est redécouverte par Arun *et al.* [AHB87], mais elle échoue dans les cas particuliers évoqués ci-dessus. Elle est améliorée par Umeyama [Ume91], et finalement par Kanatani [Kan94] qui assure que R est une rotation ($\det R = 1$) et non une réflexion ($\det R = -1$). La méthode choisie, décrite par l'Algorithme 10, est résumée en cinq étapes. Après le calcul de la matrice de corrélation entre le signal $\boldsymbol{y}$ et l'atome $\boldsymbol{\phi}$: $M_c = \boldsymbol{y}\boldsymbol{\phi}^T \in \mathbb{R}^{3\times 3}$ (étape 1), sa SVD est effectuée : $(U, \Sigma_1, V) = \text{SVD}(M_c)$ (étape 2). En définissant la matrice Σ_2 telle que (étape 3) :

$$\Sigma_2 = \begin{bmatrix} 1 & & \\ & 1 & \\ & & \det(UV^T) \end{bmatrix}, \tag{5.13}$$

la rotation optimale est : $R = U\Sigma_2 V^T$ (étape 4). La valeur de corrélation qui donne le facteur d'échelle est calculée par : $x = \text{Tr}(R\boldsymbol{\phi}\boldsymbol{y}^T) = \text{Tr}(\Sigma_2\Sigma_1) \geq 0$ (étape 5).

Cette méthode de recalage sera l'étape centrale des algorithmes de décomposition 3DRI. La démonstration du problème de Procrustes orthogonal est détaillée en Annexe 7.7.

Algorithm 10 : $(x, R) = \text{Recalage\_SVD}\ (\boldsymbol{\phi}, \boldsymbol{y})$

1: Matrice de corrélation : $M_c \leftarrow \boldsymbol{y}\boldsymbol{\phi}^T$
2: SVD : $(U, \Sigma_1, V) \leftarrow \text{SVD}(M_c)$
3: Matrice Σ_2 : $\Sigma_2 \leftarrow \text{diag}(1, 1, \det(UV^T))$
4: Matrice de rotation optimale : $R \leftarrow U\Sigma_2 V^T$
5: Valeur de corrélation : $x \leftarrow \text{Tr}(\Sigma_2\Sigma_1)$

5.3.2 Description du 3DRI-MP

Dans cette section, le 3DRI-MP qui résout le problème (5.2) est expliqué et justifié pas à pas. Un signal trivarié $\boldsymbol{y} \in \mathbb{R}^{3\times N}$ et un dictionnaire $\boldsymbol{\Psi}$ de noyaux trivariés sont considérés. Nous rappelons que le dictionnaire est normé, ce qui signifie que chaque noyau est normé. Etant donné ce dictionnaire trivarié redondant, le 3DRI-MP produit une approximation parcimonieuse du signal $\boldsymbol{y}$, comme décrit dans l'Algorithme 11.

L'initialisation (étape 1) alloue le signal étudié $\boldsymbol{y}$ au résidu $\boldsymbol{\epsilon}^0$. A l'itération courante k, l'algorithme sélectionne l'atome qui produit la plus forte décroissance de la MSE $\left\|\boldsymbol{\epsilon}^{k-1}(t)\right\|_F^2$. En notant $\boldsymbol{\epsilon}^{k-1}(t) = x_{l,\tau}R_{l,\tau}\boldsymbol{\psi}_l(t-\tau) + \boldsymbol{\epsilon}^k(t)$ et en utilisant les règles de dérivation de la trace d'une matrice [PP08], nous avons :

$$\frac{\partial \left\|\boldsymbol{\epsilon}^{k-1}(t)\right\|_F^2}{\partial x_{l,\tau}} = \frac{\partial \text{Tr}\left(\boldsymbol{\epsilon}^{k-1}(t)\boldsymbol{\epsilon}^{k-1}(t)^T\right)}{\partial x_{l,\tau}} = 2\ \text{Tr}\left(R_{l,\tau}\psi_l(t-\tau)\ \boldsymbol{\epsilon}^{k-1}(t)^T\right) = 2\ \left\langle R_{l,\tau}\boldsymbol{\psi}_l(t-\tau), \boldsymbol{\epsilon}^{k-1}(t)\right\rangle . \tag{5.14}$$

Ceci est équivalent à trouver l'atome recalé le plus corrélé au résidu $\boldsymbol{\epsilon}^{k-1}(t)$. La valeur de corrélation $x_{l,\tau}^k = \text{Tr}(R_{l,\tau}^k\ \boldsymbol{\psi}_l(t-\tau)\ \boldsymbol{\epsilon}^{k-1}(t)^T)$ est calculée pour chaque décalage temporel τ, avec $R_{l,\tau}^k$ la rotation optimale pour recaler $\boldsymbol{\psi}_l(t-\tau)$ sur $\boldsymbol{\epsilon}^{k-1}(t)$. Pour effectuer cette étape, l'algorithme Recalage_SVD est appliqué pour chaque τ et chaque $l = 1..L$ (étape 5) et ensuite, le maximum des valeurs $x_{l,\tau}^k (\geq 0)$ est recherché pour sélectionner l'atome optimal (étape 7), qui est caractérisé par son indice de noyau l^k et sa position τ^k. Les atomes sélectionnés forment un dictionnaire actif. Le vecteur x accumule les coefficients actifs qui sont les valeurs de corrélation maximales (étape 8). Les matrices de rotation associées sont groupées dans $\mathcal{R}$ (étape 9) et le résidu courant est calculé (étape 10).

Différents critères d'arrêt (étape 12) peuvent être utilisés : un seuil sur le nombre d'itérations k ou un seuil sur la rRMSE $\left\|\boldsymbol{\epsilon}^k\right\|_F / \left\|\boldsymbol{y}\right\|_F$. A la fin, le 3DRI-MP fournit une approximation K-parcimonieuse de $\boldsymbol{y}$ en utilisant les K atomes actifs :

$$\hat{\boldsymbol{y}}^K = \sum_{k=1}^{K} x_{l^k,\tau^k}^k\ R_{l^k,\tau^k}^k\ \boldsymbol{\psi}_{l^k}(t-\tau^k)\,. \tag{5.15}$$

5.3.3 Remarques sur le 3DRI-MP

Sans considérer la non-convexité de l'algorithme, s'il n'y a pas de recouvrement entre les atomes sélectionnés, le 3DRI-MP donne la projection orthogonale du signal sur le dictionnaire actif en Eq. (5.2). Sinon, il est sous-optimal puisque les recouvrements génèrent des termes de corrélation qui ne sont pas traités par le 3DRI-MP, comme observé dans [BLM12]. La différence avec le 3DRI-OMP sera expliquée après.

Algorithm 11 : $(x, \mathcal{R}) = \text{3DRI\_MP}\ (\boldsymbol{y}, \boldsymbol{\Psi})$

initialisation : $k = 1$, résidu $\boldsymbol{\epsilon}^0 = \boldsymbol{y}$, $x = \varnothing$, $\mathcal{R} = \varnothing$
repeat
  for $l \leftarrow 1, L$ **do**
    Recalage 3D pour chaque τ :
    $(x^k_{l,\tau}, R^k_{l,\tau}) \leftarrow$ Recalage_SVD ($\boldsymbol{\psi}_l(t-\tau)$, $\boldsymbol{\epsilon}^{k-1}(t)$)
  end for
  Sélection : $(l^k, \tau^k) \leftarrow \arg\max_{l,\tau}\ x^k_{l,\tau}$
  Coefficients actifs : $x \leftarrow \left[\, x\, ;\, x_{l^k,\tau^k} \,\right]$
  Matrices actives : $\mathcal{R} \leftarrow \mathcal{R} \cup R^k_{l^k,\tau^k}$
  Résidu : $\boldsymbol{\epsilon}^k \leftarrow \boldsymbol{\epsilon}^{k-1} - x^k_{l^k,\tau^k}\, R^k_{l^k,\tau^k}\, \boldsymbol{\psi}_{l^k}(t-\tau^k)$
  $k \leftarrow k+1$
until critère d'arrêt

La description de l'Algorithme 11 est détaillée pour le rendre clair et intuitif, mais sa complexité est en $O(N^2)$. Une implémentation plus rapide est possible. En effet, chacun des neuf éléments des N matrices de corrélation $M_c(l,\tau) = \boldsymbol{\epsilon}^{k-1}(t)\ \boldsymbol{\psi}_l(t-\tau)^T$ peut d'abord être calculé par FFT en $O(N\log N)$ pour chaque τ et ensuite les N recalages 3D sont calculés en $O(N)$. La complexité résultante est en $O(N\log N)$.

Dans la lancée du chapitre précédent, l'idée intuitive pour résoudre le problème d'approximation parcimonieuse (5.2) est d'utiliser les quaternions (cf. Section 3.1.1) qui sont souvent utilisés pour traiter les rotations 3D [Han06]. Cependant, cela s'avère ne pas être possible pour résoudre ce problème. En effet, la rotation d'un atome quaternionique pur $\phi \in \mathbb{H}^N$ s'écrit :

$$\phi_r = \mathfrak{q}\, \phi\, \mathfrak{q}^* \,, \tag{5.16}$$

avec $\mathfrak{q} \in \mathbb{H}$ un quaternion unitaire effectuant la rotation de l'atome. Ce modèle est non-linéaire contrairement aux décompositions parcimonieuses. Aussi, lors d'une sélection itérative spécifiée à ce modèle, nous aurions :

$$\epsilon^{k-1} = x_m\, \mathfrak{q}_m\, \phi_m\, \mathfrak{q}_m^* + \epsilon^k = \tilde{\mathfrak{q}}_m\, \phi_m\, \tilde{\mathfrak{q}}_m^* + \epsilon^k \,. \tag{5.17}$$

En appelant J le critère $J = \left\|\epsilon^{k-1}\right\|_2^2$, la dérivée $\partial J / \partial \tilde{\mathfrak{q}}_m$ dépend encore de la valeur cherchée $\tilde{\mathfrak{q}}_m$. Ainsi, le maximum de la dérivée ne peut pas être seulement calculé avec les variables ϵ^{k-1} et ϕ_m comme c'est le cas dans le MP ou l'OMP. Par conséquent, l'étape de sélection du MP/OMP ne peut pas être généralisée au modèle quaternionique (5.16) pour résoudre le problème (5.12). Remarquons que [ZSD12] utilise les quaternions pour représenter les données comme une combinaison de rotations élémentaires apprises, et non comme une somme d'atomes capables de tourner.

5.4 OMP invariant par rotation 3D

Après avoir introduit le 3DRI-MP qui n'assure pas une projection orthogonale, nous introduisons le 3DRI-OMP qui est sa version orthogonale. Les similarités et les différences sont d'abord expliquées, et l'algorithme est ensuite détaillé.

5.4.1 Description du 3DRI-OMP

Dans le 3DRI-MP décrit dans l'Algorithme 11, le coefficient de décomposition $x^k_{l^k,\tau^k}$ de l'atome sélectionné $\boldsymbol{\psi}_{l^k}(t-\tau^k)$ est la valeur de corrélation maximale (étape 8), et de même pour la matrice de rotation $R^k_{l^k,\tau^k}$ (étape 9). Il fournit une solution approchée de l'Eq. (5.2), mais elle n'est pas optimale au sens des moindres carrés étant donné qu'elle ne prend pas en compte les termes de corrélation causés par les recouvrements entre les atomes sélectionnés.
Pour améliorer les performances, nous proposons le 3DRI-OMP dans lequel les coefficients et les matrices de rotation sont calculés par projection orthogonale du signal $\boldsymbol{y}$ sur les atomes sélectionnés du dictionnaire actif. Les valeurs des coefficients et des matrices précédemment sélectionnées doivent être mises à jour pour prendre en compte le nouvel atome et donner la solution des moindres carrés de l'Eq. (5.2). Ils ne sont modifiés que s'il y a des corrélations entre le nouvel atome et les anciens. La différence entre le 3DRI-MP et le 3DRI-OMP est équivalente à celle entre le MP et l'OMP (cf. Section 1.2.2).

Le 3DRI-OMP résout les moindres carrés de l'Eq. (5.2) séquentiellement, en augmentant la valeur K itérativement. Le 3DRI-OMP décrit dans l'Algorithme 12 est similaire au 3DRI-MP pour les étapes 1 à 10, mais calcule les solutions des moindres carrés x et $\mathcal{R}$ à chaque itération k dans l'étape 11. Cela permet une meilleure sélection à l'itération suivante $k+1$. Par la suite, nous omettons l'exposant k sur les variables x_{l^k,τ^k} et R_{l^k,τ^k} afin d'alléger les notations, et un indice $\kappa = 1..k$ numérote les différents éléments qui sont déjà sélectionnés à l'itération k. Dans le 3DRI-OMP, à l'itération courante k,

- les coefficients actifs $x = \{x_{l^\kappa,\tau^\kappa}\}_{\kappa=1}^k$
- et les matrices actives $\mathcal{R} = \{R_{l^\kappa,\tau^\kappa}\}_{\kappa=1}^k$

sont les solutions de (P_k) :

$$\min_{x,\mathcal{R}} \left\| \boldsymbol{y}(t) - \sum_{\kappa=1}^{k} x_{l^\kappa,\tau^\kappa}\, R_{l^\kappa,\tau^\kappa}\, \boldsymbol{\psi}_{l^\kappa}(t-\tau^\kappa) \right\|_F^2 \quad \text{t.q.} \quad \forall \kappa \in \mathbb{N}_k,\ R_{l^\kappa,\tau^\kappa} R_{l^\kappa,\tau^\kappa}^T = Id\,. \tag{P_k}$$

La procédure d'optimisation (étape 11) qui résout ce problème est détaillée dans le paragraphe suivant.

Les critères d'arrêt (étape 14) sont les mêmes que ceux du 3DRI-MP. Finalement, le 3DRI-OMP donne une approximation K-parcimonieuse en utilisant les coefficients et les matrices des moindres carrés.

5.4.2 Procédure d'optimisation pour les coefficients et les matrices

Dans ce paragraphe, nous détaillons la procédure d'optimisation pour résoudre (P_k) à l'étape 11 de l'Algorithme 12. Le problème (P_k) est résolu par des mises à jour alternées sur les coefficients x et sur les matrices de rotation $\mathcal{R}$. De plus, chaque mise à jour est basée sur une descente de gradient.

La solution du 3DRI-MP est une bonne initialisation pour cette optimisation, étant donné qu'elle n'est pas loin de la solution optimale de (P_k). Ainsi, la procédure utilise la solution du 3DRI-MP donnée dans les étapes 8, 9 et 10 comme les valeurs initiales de x, $\mathcal{R}$ et $\boldsymbol{\epsilon}^k$. Les deux mises à jour sont maintenant détaillées. Par la suite, l'exposant j est ajouté aux variables pour numéroter l'itération courante de la procédure d'optimisation.

Algorithm 12 : $(x, \mathcal{R}) = \text{3DRI\_OMP}\ (\boldsymbol{y}, \boldsymbol{\Psi})$

initialisation : $k = 1$, résidu $\epsilon^0 = y$, $x = \varnothing$, $\mathcal{R} = \varnothing$
repeat
  for $l \leftarrow 1, L$ **do**
    Recalage 3D pour chaque τ :
    $(x_{l,\tau}, R_{l,\tau}) \leftarrow$ Recalage_SVD ($\boldsymbol{\psi}_l(t-\tau)$, $\boldsymbol{\epsilon}^{k-1}(t)$)
  end for
  Sélection : $(l^k, \tau^k) \leftarrow \arg\max_{l,\tau}\ x_{l,\tau}$
  Coefficients actifs : $x \leftarrow \left[\, x\, ;\, x_{l^k,\tau^k} \,\right]$
  Matrices actives : $\mathcal{R} \leftarrow \mathcal{R} \cup R_{l^k,\tau^k}$
  Résidu : $\epsilon^k \leftarrow \epsilon^{k-1} - x_{l^k,\tau^k}\, R_{l^k,\tau^k}\, \boldsymbol{\psi}_{l^k}(t-\tau^k)$
  Procédure d'optimisation : $(x, \mathcal{R}) \leftarrow \arg\min_{x,\mathcal{R}}\ (P_k)$
  Résidu : $\boldsymbol{\epsilon}^k \leftarrow \boldsymbol{y} - \hat{\boldsymbol{y}}^k$
  $k \leftarrow k + 1$
until critère d'arrêt

Mise à jour des coefficients x

Nous dérivons le critère (P_k) par rapport à x_{l^κ,τ^κ} pour obtenir une descente de gradient de type LMS [WH60, Wid71]. Chaque coefficient est mis à jour tel que :

$$x^j_{l^\kappa,\tau^\kappa} = x^{j-1}_{l^\kappa,\tau^\kappa} + \lambda_1^j \cdot \mathrm{Tr}\Big(R^{j-1}_{l^\kappa,\tau^\kappa}\, \boldsymbol{\psi}_{l^\kappa}(t-\tau^\kappa)\, \boldsymbol{\epsilon}^k(t)^T \Big)\,, \tag{5.18}$$

avec λ_1^j le pas de descente adaptatif. Pour nos expériences, il sera réglé tel que $\lambda_1^j = 1/j^{0.6}$. Résoudre la mise à jour des coefficients x avec une méthode de gradient est mieux que de donner la solution optimale par pseudo-inverse. En effet, si nous résolvons la mise à jour des coefficients de façon trop exacte par rapport à celles des matrices, il y a un risque que l'optimisation reste bloquée dans un minimum local.

Mise à jour des matrices de rotation $\mathcal{R}$

Cette mise à jour doit maintenir l'orthogonalité des matrices de rotation $\mathcal{R}$. Comme précédemment, le LMS peut être utilisé pour les matrices donnant une mise à jour additive. Pour garantir l'orthogonalité des matrices mises à jour, une deuxième étape de pénalisation doit être rajoutée [Plu05]. Les désavantages sont que cette mise à jour est empiriquement moins robuste au bruit et qu'elle est effectuée en deux étapes au lieu d'une. C'est pourquoi nous choisissons une mise à jour multiplicative qui garde intrinsèquement la matrice de rotation orthogonale dans la variété de Stiefel orthogonale [NA05, Plu05]. Chaque matrice de rotation R_{l^κ,τ^κ} est mise à jour en suivant la géodésique de la variété :

$$R^j_{l^\kappa,v\tau^\kappa} = \mathrm{expm}(-\mu G^{j-1}_\kappa)\, R^{j-1}_{l^\kappa,\tau^\kappa}\,, \tag{5.19}$$

$$\text{avec}\quad G^{j-1}_\kappa = x^j_{l^\kappa,\tau^\kappa} \left(R^{j-1}_{l^\kappa,\tau^\kappa}\, \boldsymbol{\psi}_{l^\kappa}(t-\tau^\kappa)\, \boldsymbol{\epsilon}^k(t)^T - \boldsymbol{\epsilon}^k(t)\, \boldsymbol{\psi}_{l^\kappa}(t-\tau^\kappa)^T\, {R^{j-1}_{l^\kappa,\tau^\kappa}}^T \right). \tag{5.20}$$

La constante μ est fixée à 0.1 pour les données traitées par la suite et expm est la fonction exponentielle de matrice.

Optimisation alternée

La procédure d'optimisation qui résout (P_k) à l'étape 11 est résumée :

1 : **initialisation** de x^0, $\mathcal{R}^0$ et $\boldsymbol{\epsilon}^k$ sur la solution du 3DRI-MP
2 : **for** $j \leftarrow 1, J$ **do**
3 : **for** $\kappa \leftarrow 1, k$ **do**
4 : Mise à jour du coefficient $x^j_{l^\kappa,\tau^\kappa}$ par Eq. (5.18)
5 : **end for**
6 : Mise à jour du résidu $\boldsymbol{\epsilon}^k$
7 : **for** $\kappa \leftarrow 1, k$ **do**
8 : Mise à jour de la matrice $R^j_{l^\kappa,\tau^\kappa}$ par Eq. (5.19) et mise à jour du résidu $\boldsymbol{\epsilon}^k$
9 : **end for**
10 : **end for** .

Le nombre d'itération J peut être choisi constant, ou fonction de k (de cette façon, les derniers coefficients ont plus d'itérations pour approcher la solution des moindres carrés que les premiers). Remarquons qu'il est inutile d'effectuer la procédure d'optimisation à la première itération $k = 1$, étant donné qu'il n'y a pas de recouvrements dans le dictionnaire actif réduit à un seul atome.

A cause de la non-convexité des mises à jour alternées, même avec assez d'itérations, cette procédure d'optimisation n'est pas garantie de converger vers la solution optimale des moindres carrés de (P_k).

5.4.3 Remarques sur le 3DRI-OMP

Le premier élément à noter est que le 3DRI-OMP n'a pas une unique implémentation. Ici, nous avons présenté une implémentation possible pour la procédure d'optimisation de (P_k). Cependant, d'autres choix peuvent être explorés pour améliorer cette optimisation alternée des moindres carrés.

Remarquons que le 3DRI-MP peut être vu comme un 3DRI-OMP sans optimisation du problème (P_k), *i.e.* en choisissant $J = 0$. Cela explique pourquoi le 3DRI-MP est plus rapide mais sous-optimal.

Dans ces deux dernières sections, nous avons présenté les algorithmes 3DRI-MP et 3DRI-OMP traitant le problème d'approximation parcimonieuse (5.2). Cependant, les décompositions ne sont efficaces que si le dictionnaire est adapté.

5.5 Etat de l'art des apprentissages 3D

Nous voulons maintenant mettre en place un algorithme d'apprentissage de dictionnaire adapté au modèle 3DRI (5.1). Dans cette section, nous passons en revue les méthodes qui apprennent des motifs à partir de données tricomposantes.

5.5.1 Apprentissage de base

Les méthodes d'apprentissage de base utilisent l'algorithme EM (cf. Section 1.3.3), qui alterne entre l'estimation des coefficients et celle des vecteurs de la base. Dans [BHB00], Bregler *et al.* décrivent un objet 3D comme une combinaison linéaire de vecteurs de forme. Comme ils sont spécifiques à l'objet étudié, ils doivent être estimés à partir des données. Torresani *et al.* [THB04] utilisent un algorithme EM

généralisé pour apprendre des formes 3D qui sont adaptées aux données étudiées, et cette approche a été améliorée par [THB08]. Dans [ASKK11], Akhter *et al.* représentent un objet 3D comme une combinaison linéaire de vecteurs de trajectoire dans le modèle dual (5.3). Comme les vecteurs sont indépendants de l'objet étudié, des bases génériques comme la DCT peuvent être utilisées [ASKK11]. Cependant, Su *et al.* [SLY10] utilisent un algorithme de type EM pour apprendre des trajectoires unicomposantes adaptées aux données, et ils obtiennent de meilleurs résultats que [THB04] et [ASKK11].

5.5.2 Apprentissage de dictionnaire

L'apprentissage de dictionnaire, présenté en Section 1.3, apprend un dictionnaire redondant au lieu d'une base et cela permet une plus grande flexibilité de représentation. Les avantages des dictionnaires sur les bases sont détaillés en Section 1.1.2.

Considérant un ensemble de signaux trivariés représentant des objets 3D instantanés, Zhang *et al.* apprennent un dictionnaire de formes [ZZZ+12], mais sans dimension temporelle. Etudiant un ensemble de signaux temporels trivariés, le M-DLA apprend un dictionnaire trivarié basé sur le modèle (5.4) (cf. Section 2.3), mais sans rotation entre les trois composantes. Par la suite, nous voulons mettre en place le *3D Rotation Invariant* DLA (3DRI-DLA) qui apprend un dictionnaire trivarié capable de tourner grâce au modèle (5.5).

5.6 DLA invariant par rotation 3D

Dans cette section, nous expliquons comment estimer un dictionnaire 3DRI à partir d'un ensemble d'apprentissage trivarié.

5.6.1 Description du 3DRI-DLA

Un ensemble d'apprentissage de signaux trivariés $\boldsymbol{Y} = \{\boldsymbol{y}_p\}_{p=1}^{P}$ est considéré, avec l'indice p ajouté à toutes les variables. Le 3DRI-DLA décrit dans l'Algorithme 13 est de nature *online* (cf. Section 1.3.2). Il s'agit donc d'une alternance entre une approximation parcimonieuse 3DRI et une mise à jour du dictionnaire 3DRI. Dans sa structure, il est proche du M-DLA décrit en Section 2.3.

L'approximation parcimonieuse 3DRI de chaque signal d'apprentissage $\boldsymbol{y}_p$ est réalisée par l'algorithme de décomposition 3DRI-OMP (étapes 4-5) :

$$x_p, \mathcal{R}_p = \arg\min_{x,\mathcal{R}} \left\| \boldsymbol{y}_p(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau}\, R_{l,\tau}\, \boldsymbol{\psi}_l(t-\tau) \right\|_F^2$$

$$\text{t.q.} \quad \|x\|_0 \leq K \quad \text{et} \quad \forall l \in \mathbb{N}_L, \forall \tau \in \sigma_l,\ R_{l,\tau} R_{l,\tau}^T = Id\,, \tag{5.21}$$

suivie de la mise à jour du dictionnaire (étapes 6-7) :

$$\boldsymbol{\Psi} = \arg\min_{\boldsymbol{\Psi}} \left\| \boldsymbol{y}_p(t) - \sum_{l=1}^{L} \sum_{\tau \in \sigma_l} x_{l,\tau;p}\, R_{l,\tau;p}\, \boldsymbol{\psi}_l(t-\tau) \right\|_F^2 \text{ t.q. } \forall\, l \in \mathbb{N}_L,\ \|\boldsymbol{\psi}_l\|_F = 1\,. \tag{5.22}$$

Pour effectuer cette optimisation, nous utilisons une descente de gradient stochastique (SGD) par le LMS. Comme il a la particularité de traiter des matrices de rotation 3D, nous l'appelons 3DRI-LMS et

il est dérivé en utilisant les règles de dérivation de [PP08] :

$$-\frac{\partial \parallel \boldsymbol{\epsilon}_p(t) \parallel_F^2}{\partial \boldsymbol{\psi}_l} = 2 \sum_{\tau \in \sigma_l} x_{l,\tau;p} \, R_{l,\tau;p}^T \, \underline{\boldsymbol{\epsilon}}_\tau(t) \, , \tag{5.23}$$

avec $\underline{\boldsymbol{\epsilon}}_\tau$ représentant l'erreur localisée au décalage τ et restreinte au support temporel de $\boldsymbol{\psi}_l$, *i.e.* $\underline{\boldsymbol{\epsilon}}_\tau = \boldsymbol{\epsilon}|_{t=\tau..\tau+T_l}$, avec T_l la longueur de $\boldsymbol{\psi}_l$. L'itération courante étant notée i, pour $l = 1..L$, chaque noyau trivarié $\boldsymbol{\psi}_l$ est mis à jour tel que :

$$\boldsymbol{\psi}_l^i(\underline{t}) = \boldsymbol{\psi}_l^{i-1}(\underline{t}) + \lambda_2^i \cdot \sum_{\tau \in \sigma_l} x_{l,\tau;p}^i \, R_{l,\tau;p}^i{}^T \, \boldsymbol{\epsilon}_p^{i-1}(\underline{t} + \tau) \, , \tag{5.24}$$

avec $\underline{t}$ les indices d'échantillons limités au support temporel de $\boldsymbol{\psi}_l$ et λ_2 le pas de descente adaptatif. Il est choisi tel que $\lambda_2^i = 1/i$. Les trois composantes du noyau trivarié $\boldsymbol{\psi}_l$ sont mises à jour simultanément. Les noyaux sont normalisés à la fin de chaque itération et leurs longueurs peuvent être modifiées en fonction de la quantité d'énergie présente sur leurs bords, comme dans le M-DLA.

Au début de l'algorithme (étape 1), les noyaux sont initialisés sur du bruit blanc uniforme (entre 0 et 1) et sont ensuite normalisés. A la fin (étape 8), différents critères d'arrêt peuvent être utilisés : un seuillage sur le nombre d'itérations ou sur la rRMSE calculée sur l'ensemble d'apprentissage.

Algorithm 13 : $\boldsymbol{\Psi}$ = 3DRI_DLA $(\{\boldsymbol{y}_p\}_{p=1}^P)$

1: **initialisation** : $i = 1, \boldsymbol{\Psi}^0 = \{L$ noyaux de bruit blanc$\}$
2: **repeat**
3: **for** $p \leftarrow 1, P$ **do**
4: Approximation parcimonieuse 3DRI : $(x_p^i, \mathcal{R}_p^i) \leftarrow$ 3DRI-OMP $(\boldsymbol{y}_p, \boldsymbol{\Psi}^{i-1})$
5: Mise à jour du dictionnaire : $\boldsymbol{\Psi}^i \leftarrow$ 3DRI-LMS $(\boldsymbol{y}_p, x_p^i, \mathcal{R}_p^i, \boldsymbol{\Psi}^{i-1})$
6: $i \leftarrow i + 1$
7: **end for**
8: **until** critère d'arrêt

5.6.2 Remarques sur le 3DRI-DLA

Durant l'apprentissage, nous pouvons observer que certains noyaux ne convergent pas, et ils ne sont pas utilisés dans les décompositions étant donné qu'ils ressemblent à du bruit blanc. Par conséquent, ils sont supprimés du dictionnaire à la fin de l'apprentissage.

Dans le 3DRI-DLA, le 3DRI-OMP est arrêté par un seuil sur le nombre d'itérations. La rRMSE ne peut pas être utilisée ici car, au début de l'apprentissage, les noyaux de bruit blanc ne peuvent pas engendrer un pourcentage donné de l'espace étudié. De plus, à la première itération du 3DRI-DLA ($i = 1$), l'optimisation de (P_k) du 3DRI-OMP (étape 11) n'est pas effectuée. Ainsi, la constante est fixée à : $J = i - 1$.

5.7 Expériences sur données simulées

Dans cette section, les méthodes introduites sont appliquées à des données simulées et comparées afin d'évaluer leurs performances.

5.7.1 Décompositions invariantes par rotation 3D

La première expérience compare les performances du 3DRI-MP et du 3DRI-OMP. Un dictionnaire $\mathbf{\Psi}$ de $L = 50$ noyaux trivariés est créé aléatoirement : les noyaux normalisés sont générés comme des bruits blancs Gaussiens. La longueur des noyaux est de $T = 65$ échantillons. Cent signaux de $N = 250$ échantillons sont composés de la somme de $K = 15$ atomes, pour lesquels les coefficients (strictement positifs), les matrices de rotation, les indices de noyaux, et les paramètres de décalage sont tirés selon des distributions uniformes. En conséquence, nous avons des recouvrements entre les différents atomes. L'approximation de chaque signal est effectuée par les algorithmes 3DRI-MP et 3DRI-OMP avec $K = 15$ itérations, afin de retrouver les 15 atomes simulés. Le M-OMP utilisé dans un cas trivarié est aussi testé sur ces signaux pour se comparer au modèle trivarié (5.4). Un dictionnaire univarié normalisé est calculé en moyennant les 3 composantes de celui trivarié, et le *multichannel* OMP (Mch-OMP) [GRSV07] utilisé pour se comparer au modèle tricanal de Akhter *et al.* (5.3). La rRMSE $\left\|\boldsymbol{\epsilon}^k\right\|_F / \|\boldsymbol{y}\|_F$ est moyennée (moyenne et écart-type) sur les 100 signaux et est tracée en Fig. 5.3 en fonction des itérations internes $k = 1..K$ des quatre algorithmes.

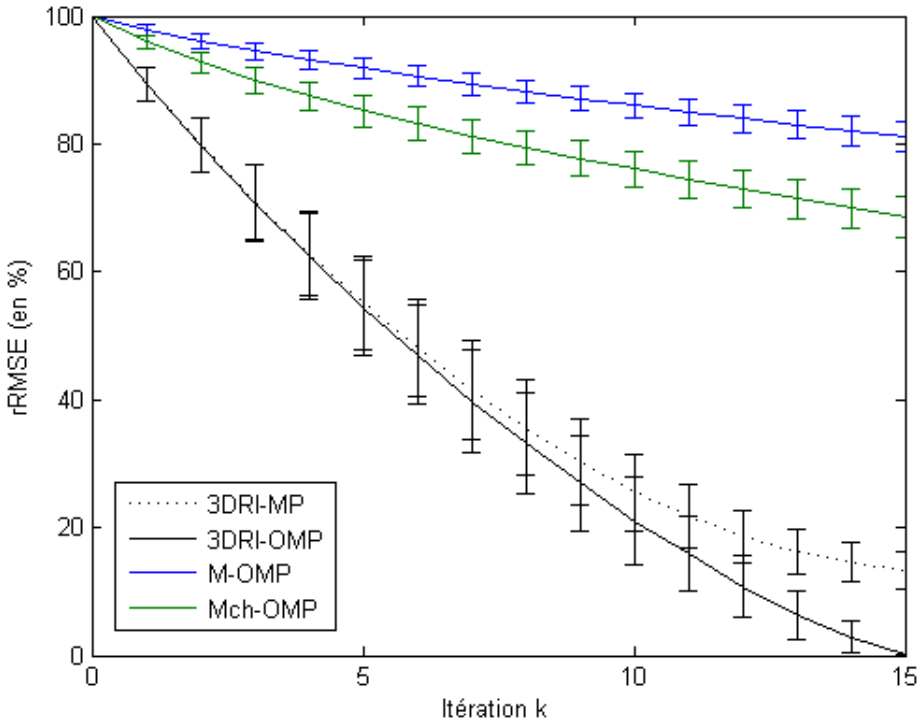


Fig. 5.3 – Comparaison entre les performances des 3DRI-MP, 3DRI-OMP, M-OMP et Mch-OMP. La rRMSE moyennée sur les 100 signaux est tracée en fonction de l'itération interne k.

Nous observons que le 3DRI-OMP donne de meilleures performances d'approximation que le 3DRI-MP. Au début, les deux algorithmes ont des comportements similaires étant donné qu'il n'y a pas beaucoup d'atomes actifs dans les décompositions, et donc que la procédure d'optimisation (cf. Section 5.4.2) du 3DRI-OMP n'a pas d'impact significatif. A la fin des K itérations, le 3DRI-OMP a une rRMSE de 0.2%, alors que le 3DRI-MP a une rRMSE de 13.4%. Cela montre l'optimalité du 3DRI-OMP, qui fournit la solution des moindres carrés, contrairement au 3DRI-MP. Parfois, les recouvrements entre atomes sont

importants, et le 3DRI-OMP n'identifie pas les bons indices de noyaux $\{l^k\}_{k=1}^K$ et les bonnes positions temporelles $\{\tau^k\}_{k=1}^K$. Cela explique que la rRMSE moyenne de la Fig. 5.3 n'est pas exactement nulle à la fin du 3DRI-OMP ($k = 15$). Concernant le M-OMP et le Mch-OMP, leurs rRMSE sont élevées car leurs dictionnaires ne sont pas adaptés aux données tournées. Cependant, le Mch-OMP est meilleur car il possède trois degrés de liberté (coefficients) au lieu d'un seul pour le M-OMP.

Cette expérience montre la pertinence des algorithmes 3DRI pour la décomposition de données tournées, ainsi que l'optimalité du 3DRI-OMP sur le 3DRI-MP due à la projection orthogonale. Après avoir testé les décompositions, nous allons tester les apprentissages.

5.7.2 Apprentissages de dictionnaires invariants par rotation 3D

Cette expérience a pour but de tester les capacités d'apprentissage du 3DRI-DLA. Le protocole expérimental est proche de celui de l'expérience effectuée en Section 2.4. Un dictionnaire $\boldsymbol{\Psi}$ de $L = 45$ noyaux trivariés normalisés est créé à partir de bruit blanc uniforme, et la longueur des noyaux est $T = 18$ échantillons. L'ensemble d'apprentissage $\boldsymbol{Y}_1$ est composé de $P = 2000$ signaux de longueur $N = 20$, et il est généré à partir de ce dictionnaire. Les décalages circulaires ne sont pas permis pour les noyaux, ce qui laisse trois décalages possibles. Chaque signal d'apprentissage est composé de la somme de trois atomes dont les coefficients (strictement positifs), les matrices de rotation, les indices de noyau et les paramètres de décalages sont tirés aléatoirement. Un bruit blanc Gaussien est aussi ajouté à plusieurs RSB : 10, 20 et 30 dB, et sans bruit. L'initialisation du dictionnaire est effectuée sur l'ensemble d'apprentissage, et le dictionnaire appris $\hat{\boldsymbol{\Psi}}$ est obtenu après 80 itérations sur l'ensemble d'apprentissage (*i.e.* $i \leq 80 \times P$). Pour les 40 premières itérations, le pas adaptatif est choisi tel que : $\lambda_2^i = 1/(i-q+1)^{1.5}$, *i.e.* constant pour chaque boucle sur l'ensemble d'apprentissage, et il est ensuite gardé constant pour les dernières itérations : $\lambda_2^i = 1/40^{1.5}$. Le protocole expérimental est légèrement changé pour donner l'ensemble d'apprentissage $\boldsymbol{Y}_2$. Dans ce cas, il n'y a plus de rotation lors de la fabrication des signaux d'apprentissage avec les noyaux trivariés (*i.e.* $R_{l,\tau} = Id$).

Dans l'expérience, un noyau appris $\hat{\boldsymbol{\psi}}_l$ est considéré comme retrouvé si son produit scalaire c_l, après recalage 3D, avec son noyau original correspondant est tel que :

$$c_l \geq 0.99 \quad \text{avec} \quad (c_l, .) = \text{Recalage\_SVD}(\hat{\boldsymbol{\psi}}_l, \boldsymbol{\psi}_l)\,. \tag{5.25}$$

Comme le M-DLA utilisé dans un cas trivarié est aussi testé, sa condition de détection est simplement :

$$c_l = \left|\left\langle \boldsymbol{\psi}_l, \hat{\boldsymbol{\psi}}_l \right\rangle\right| \geq 0.99\,. \tag{5.26}$$

La Table 5.1 résume les taux de détection moyennés sur 10 apprentissages, donnés en pourcentage en fonction du niveau de bruit (colonnes) et du type d'apprentissage (lignes). 3DRI-DLA (a) donne les résultats pour l'ensemble d'apprentissage $\boldsymbol{Y}_1$ et la condition de détection (5.25) ; 3DRI-DLA (b) donne les résultats pour l'ensemble d'apprentissage $\boldsymbol{Y}_2$ et la condition de détection (5.25) ; et M-DLA donne les résultats pour l'ensemble d'apprentissage $\boldsymbol{Y}_2$ et la condition de détection (5.26).

La même expérience est faite en Section 2.4, mais dans le cas univarié. Tout d'abord, nous remarquons que les résultats du M-DLA dans le cas trivarié (ligne 4) sont similaires à ceux du cas univarié (cf. Fig. 2.4). Ainsi, la nature trivariée des signaux n'a pas d'influence sur les résultats considérés. Cependant, les différences dans les résultats des trois lignes de la Table 5.1 peuvent sembler surprenantes.

TABLE 5.1 – Taux de détection (en %).

Niveau de bruit (en dB)	10	20	30	Pas de bruit
3DRI-DLA (a)	96.8	95.8	95.1	96.8
3DRI-DLA (b)	58.0	72.2	76.3	77.8
M-DLA	56.4	57.3	61.1	61.1

D'une part, les noyaux trivariés sont de bruit blanc mais sont cependant corrélés car ils sont courts ($T = 18$). Pour 100 000 essais, les corrélations moyennes entre deux bruits blancs Gaussiens trivariés normalisés sont de 0.2 ; et si nous procédons en plus à un recalage 3D, elles sont de 0.4. Ainsi, dans ce cas, le recalage 3D double la valeur de la corrélation. L'approche 3DRI améliore intrinsèquement la capacité à retrouver les noyaux ce qui explique les meilleurs résultats du 3DRI-DLA sur le M-DLA utilisé dans le cas trivarié (ligne 3 et 4 de la Table 5.1).
D'autre part, les rotations aléatoires dans l'ensemble d'apprentissage $\boldsymbol{Y}_1$ fournissent une convergence plus rapide et plus stable du 3DRI-DLA non-convexe comparé à $\boldsymbol{Y}_2$. Quand ce qui peut varier varie (les matrices de rotation), il est plus facile d'identifier ce qui ne tourne pas (les noyaux). Comme les noyaux sont tournés dans de multiples orientations dans $\boldsymbol{Y}_1$, il est facile de les apprendre car ils sont les invariants rotationnels des données. Ce phénomène explique pourquoi les résultats de 3DRI-DLA(a) sont meilleurs que ceux de 3DRI-DLA(b) (qui est capable de donner des résultats similaires avec plus d'itérations).

Pour conclure cette section, les algorithmes proposés ont été évalués et comparés avec succès sur des données simulées.

5.8 Expériences sur données réelles

Dans cette section, nos méthodes sont appliquées à des données réelles de Langage Parlé Complété (LPC) français. Les données sont d'abord présentées et nos méthodes sont ensuite appliquées sur les données originales et sur les données tournées.

5.8.1 Données d'application

Nos méthodes sont appliquées à des mouvements de LPC français, qui est un langage gestuel complétant la lecture labiale [GBB+05, Gib06]. Il sert à lever les ambiguïtés de lecture labiale, comme par exemple entre les prononciations de *pain* et *bain*. Ce langage associe des articulations de la bouche à des gestes de la main. Dans la suite, seulement les mouvements de la main sont étudiés. La main peut être disposée dans des formes différentes (configurations des doigts correspondant aux consonnes) comme montré en Fig. 5.4, et dans des placements différents (localisations sur le visage correspondant aux voyelles).

Pour effectuer l'acquisition, les marqueurs rétro-réflectifs sont disposés sur la main d'une personne qualifiée pratiquant habituellement le LPC français [GBB+05, Gib06]. Les données sont acquises par 12 caméras qui enregistrent les coordonnées 3D des marqueurs en utilisant un Système de Capture de Mouvement Vicon®, comme montré en Fig. 5.5. A la fin de l'acquisition, les coordonnées tricomposantes

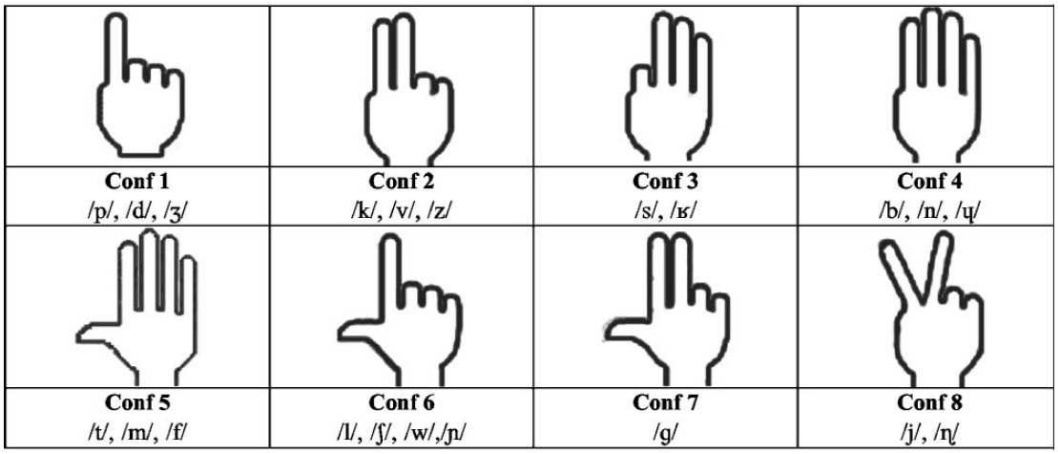


Fig. 5.4 – Formes de la main du Langage Parlé Complété français (recopié de [GBB+05]).

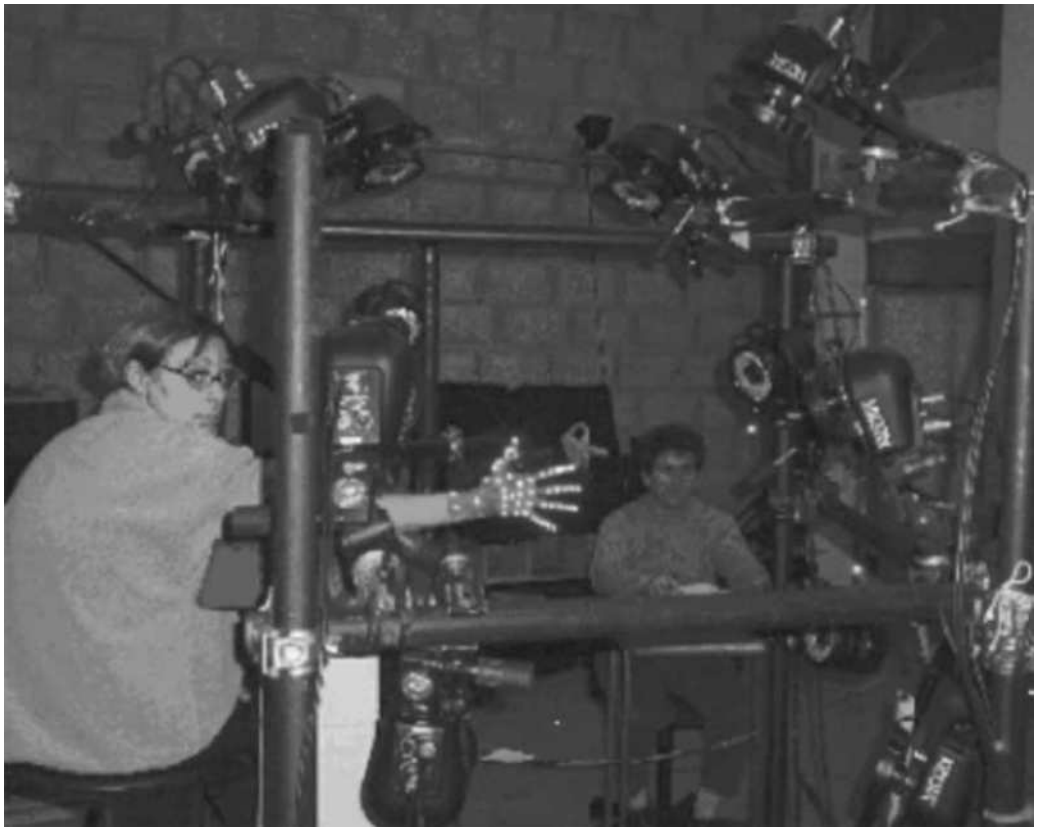


Fig. 5.5 – Acquisition des données de mouvement : les positions 3D des marqueurs rétro-réflectifs sont données par un Système de Capture de Mouvement Vicon® (recopié de [GBB+05]).

sont obtenues pour chaque marqueur à une fréquence d'échantillonnage de 120 Hz. Ces données brutes sont découpées en $P = 57$ signaux de positions, et sont dérivées pour donner les signaux de vitesse $\mathfrak{v}_x$, $\mathfrak{v}_y$ et $\mathfrak{v}_z$. Ces signaux $\mathfrak{v} = [\, \mathfrak{v}_x \,;\, \mathfrak{v}_y \,;\, \mathfrak{v}_z \,]$ sont les données d'entrée de nos algorithmes. Un seul marqueur est étudié et, parmi tous les marqueurs disponibles montrés en Fig. 5.5, nous choisissons arbitrairement

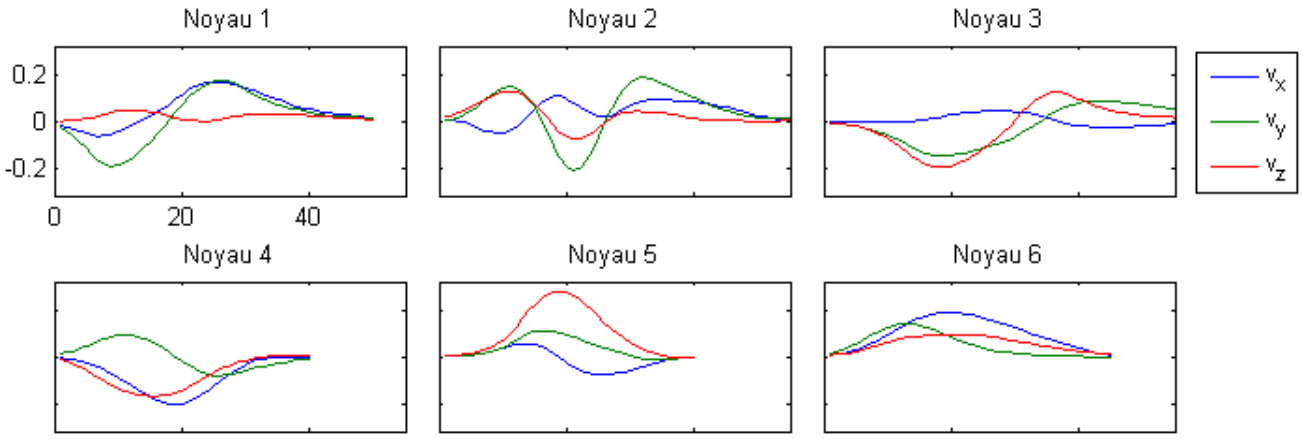


Fig. 5.6 – Dictionnaire de $L = 6$ noyaux trivariés appris par 3DRI-DLA. Chaque noyau de vitesse possède trois composantes $\mathfrak{v}_x$ (bleu), $\mathfrak{v}_y$ (vert) et $\mathfrak{v}_z$ (rouge).

celui situé sur le haut du pouce. Les unités des données sont les centimètres (cm) pour les positions et les centimètres par seconde (cm/s) pour les vitesses. Pour alléger les figures, les unités sont omises par la suite.

Remarquons que les algorithmes introduits sont indépendants du système de capture de mouvement : systèmes inertiels, magnétiques ou optiques (marqueurs passifs ou actifs). Toutes ces acquisitions de données donnent finalement des signaux spatiaux 3D, qui sont ensuite traités par nos algorithmes.

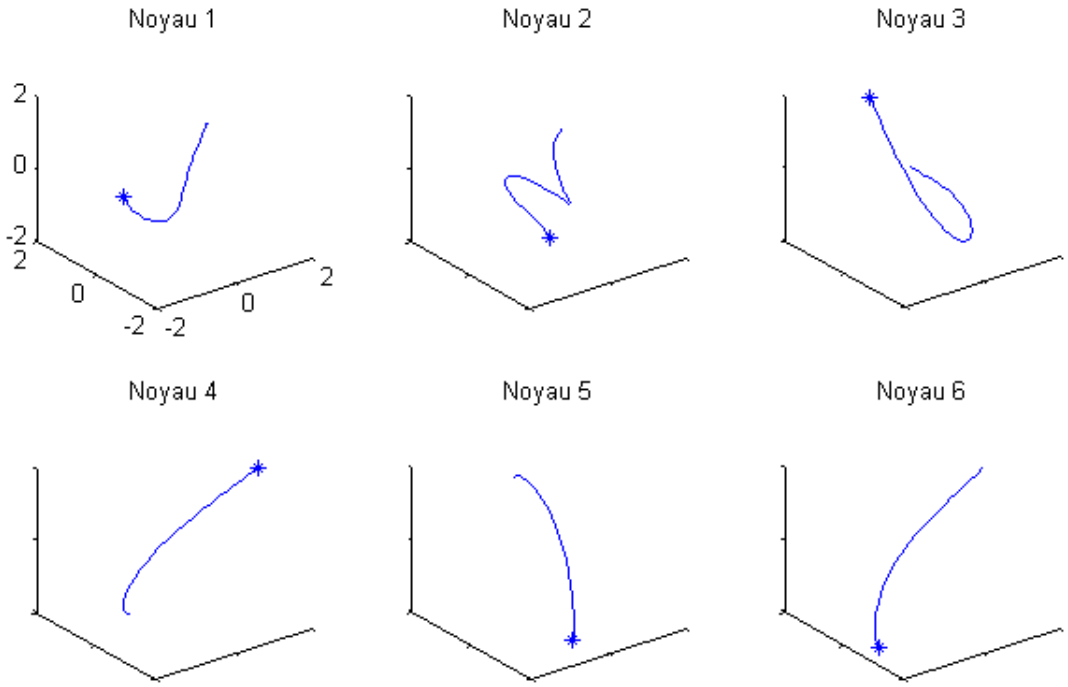


Fig. 5.7 – Dictionnaire de trajectoires 3D associé au dictionnaire NOLD appris par le 3DRI-DLA.

5.8.2 Expérience 1 : apprentissage de dictionnaires

Dans cette expérience, nous appliquons l'algorithme d'apprentissage 3DRI-DLA aux données trivariées décrites dans le paragraphe précédent. Nous faisons aussi une comparaison avec le M-DLA utilisé dans le cas trivarié. Comme en Section 4.3.2, un dictionnaire appris par le M-DLA est dit orienté et est appelé OLD, étant donné que les noyaux sont appris dans une orientation fixe, sans rotation possible. De même, un dictionnaire appris par le 3DRI-DLA est dit non-orienté et est appelé NOLD, puisque les noyaux sont invariants par rotation et peuvent être utilisés dans toutes les orientations possibles. La DCT utilisée avec le Mch-OMP (Mch-DCT) est aussi considérée afin de comparer les résultats du modèle (5.3).

Un dictionnaire NOLD est appris avec $L = 6$ noyaux trivariés, et il est affiché en Fig. 5.6. Chaque noyau de vitesse possède trois composantes $\mathfrak{v}_x$ (bleu), $\mathfrak{v}_y$ (vert) et $\mathfrak{v}_z$ (rouge). Cette convention pour les styles de la légende de la Fig. 5.6 sera conservée tout au long du chapitre. La rRMSE moyennée sur l'ensemble d'apprentissage est de 18.2% avec $K = 20$ atomes dans chaque décomposition. Les signaux de vitesse sont intégrés seulement pour fournir une représentation plus visuelle du dictionnaire, même s'il n'est pas facile de visualiser des trajectoires 3D sur une figure 2D. La Fig. 5.7 montre les trajectoires 3D capables de tourner associées au NOLD, et les étoiles marquent le point de départ des trajectoires. A cause de l'intégration, deux noyaux de vitesse différents peuvent fournir des trajectoires (noyaux intégrés) très proches. Ces trajectoires extraites par le 3DRI-DLA correspondent aux motifs élémentaires des données : elles sont les primitives du mouvement du LPC.

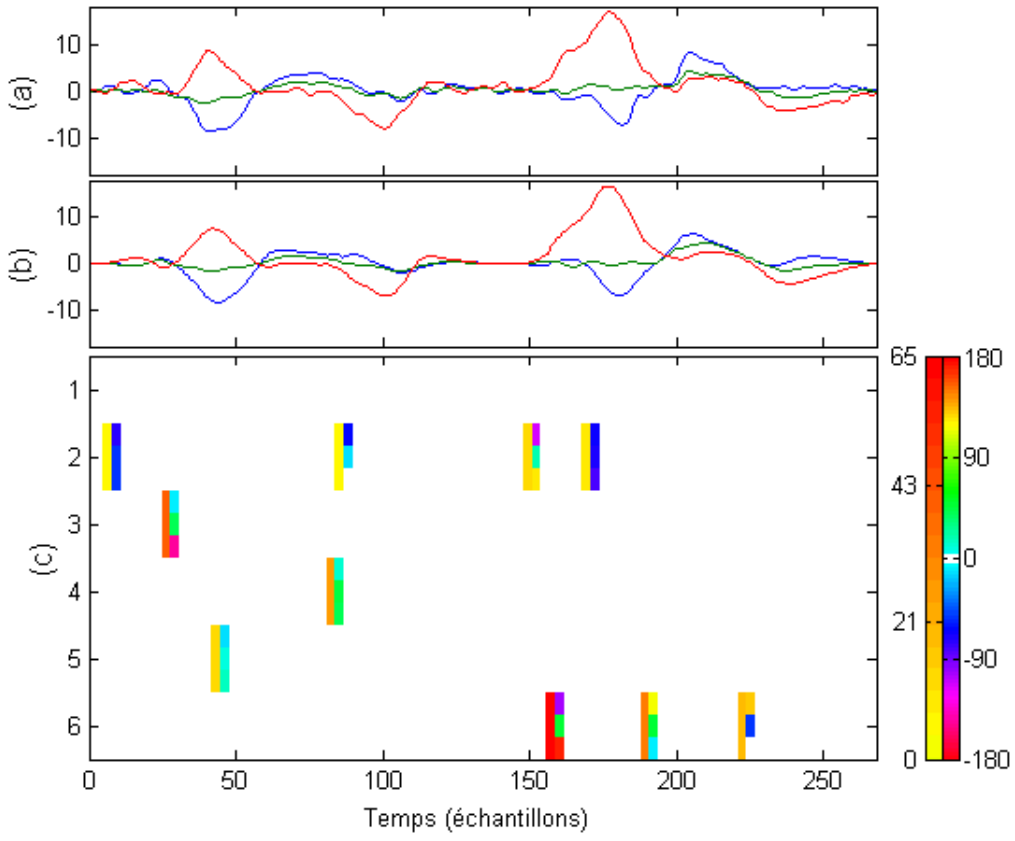


Fig. 5.8 – Signal de vitesse original $\boldsymbol{y}_{52}$ (a) et son approximation $\hat{\boldsymbol{y}}_{52}$ (b), et le spikegramme associé (c).

De la même manière, différents dictionnaires OLDs sont appris avec $L = 6$, 10 et 14 noyaux. Avec $K = 20$ atomes actifs, les rRMSE moyennées sont respectivement 27.7%, 25.7% et 24.2%. Même avec deux fois plus de noyaux que le NOLD, l'apprentissage orienté ne peut pas représenter l'espace aussi bien que l'apprentissage non-orienté. Pour le Mch-DCT, la rRMSE moyennée est de 48.1%. Ces résultats montrent déjà la pertinence de l'approche non-orientée fournissant un dictionnaire de noyaux efficace qui est plus compact que ceux orientés, eux-mêmes meilleurs que la Mch-DCT.

5.8.3 Expérience 2 : décompositions des données

Nous appliquons ensuite le 3DRI-OMP avec le NOLD, afin d'effectuer une décomposition non-orientée adaptée. Nous expliquons comment visualiser les coefficients obtenus d'une décomposition invariante par translation et par rotation 3D. Chaque atome actif possède un coefficient x_{l^k,τ^k} et une matrice de rotation R_{l^k,τ^k}, ce qui fait quatre paramètres à afficher. Pour garder une bonne visualisation, chaque matrice de rotation est transformée de manière univoque en 3 angles d'Euler $\theta^1_{l^k,\tau^k}$, $\theta^2_{l^k,\tau^k}$ et $\theta^3_{l^k,\tau^k}$ [Han06]. A partir de ces angles, nous imitons la visualisation quaternionique (cf. Section 3.2.3) qui utilise deux échelles de couleurs : une pour les amplitudes et une pour les angles d'Euler, définie de -180° à +180° et visuellement circulaire. Au final, chaque atome actif est affiché avec six indications :

- la position temporelle τ^k en abscisse,
- l'indice de noyaux l^k en ordonnée,
- l'amplitude du coefficient $x_{l^k,\tau^k} \geq 0$ en échelle de couleur,
- les 3 paramètres $\theta^1_{l^k,\tau^k}$, $\theta^2_{l^k,\tau^k}$, $\theta^3_{l^k,\tau^k}$ affichés verticalement (échelle de couleur circulaire).

Le signal $\boldsymbol{y}_{52}$ est décomposé par le 3DRI-OMP avec le NOLD et est présenté en Fig. 5.8. Le signal (a) est le signal original $\boldsymbol{y}_{52}$ composé de trois composantes, le signal (b) est l'approximation $\hat{\boldsymbol{y}}_{52}$ avec $K = 10$ atomes et le spikegramme associé est affiché en (c). Le faible nombre d'atomes utilisés dans l'approximation du signal montre la parcimonie de la décomposition. Nous rappelons que les atomes principaux sont ceux de fortes amplitudes, comme les noyaux 3, 4 et 6, et ils concentrent une grande part de l'énergie. Les atomes secondaires codent les variabilités entre les différentes réalisations du même geste.

Les trajectoires approximées du signal $\boldsymbol{y}_{52}$ avec $K = 5$ atomes sont affichées en Fig. 5.9. La trajectoire 3D originale est tracée en Fig. 5.9(a), l'approximation non-orientée en utilisant le 3DRI-OMP avec le NOLD en Fig. 5.9(b), l'approximation orientée en utilisant le M-OMP avec le OLD ($L = 6$) en Fig. 5.9(c), l'approximation Mch-DCT en Fig. 5.9(d). Sur cette figure, nous constatons la dégradation de la qualité des approximations. Pour confirmer quantitativement cette impression, nous traçons en Fig. 5.10 les différentes rRMSE en fonction de la valeur de parcimonie K. Ces résultats montrent l'efficacité du dictionnaire NOLD par rapport aux dictionnaires OLDs, eux-mêmes meilleurs que la Mch-DCT.

Maintenant, nous sommes intéressés par la contribution des atomes principaux. La trajectoire du signal $\boldsymbol{y}_{52}$ est reconstruite en Fig. 5.11 en utilisant ses atomes principaux. Par exemple, pour la reconstruction de la Fig. 5.11(a), $\boldsymbol{y}_{52}$ est reconstruit comme la somme des noyaux NOLD 3, 4 et 6 (utilisé trois fois), qui sont spécifiés par les amplitudes et les rotations du spikegramme (Fig. 5.8). Les reconstructions non-orientée et orientée sont maintenant comparées. La reconstruction n'est pas possible pour la DCT puisque ses atomes ne sont pas localisés en temps. En considérant la trajectoire 3D originale affichée en Fig. 5.9(a), la trajectoire reconstruite non-orientée est tracée en Fig. 5.11(a) et celle orientée

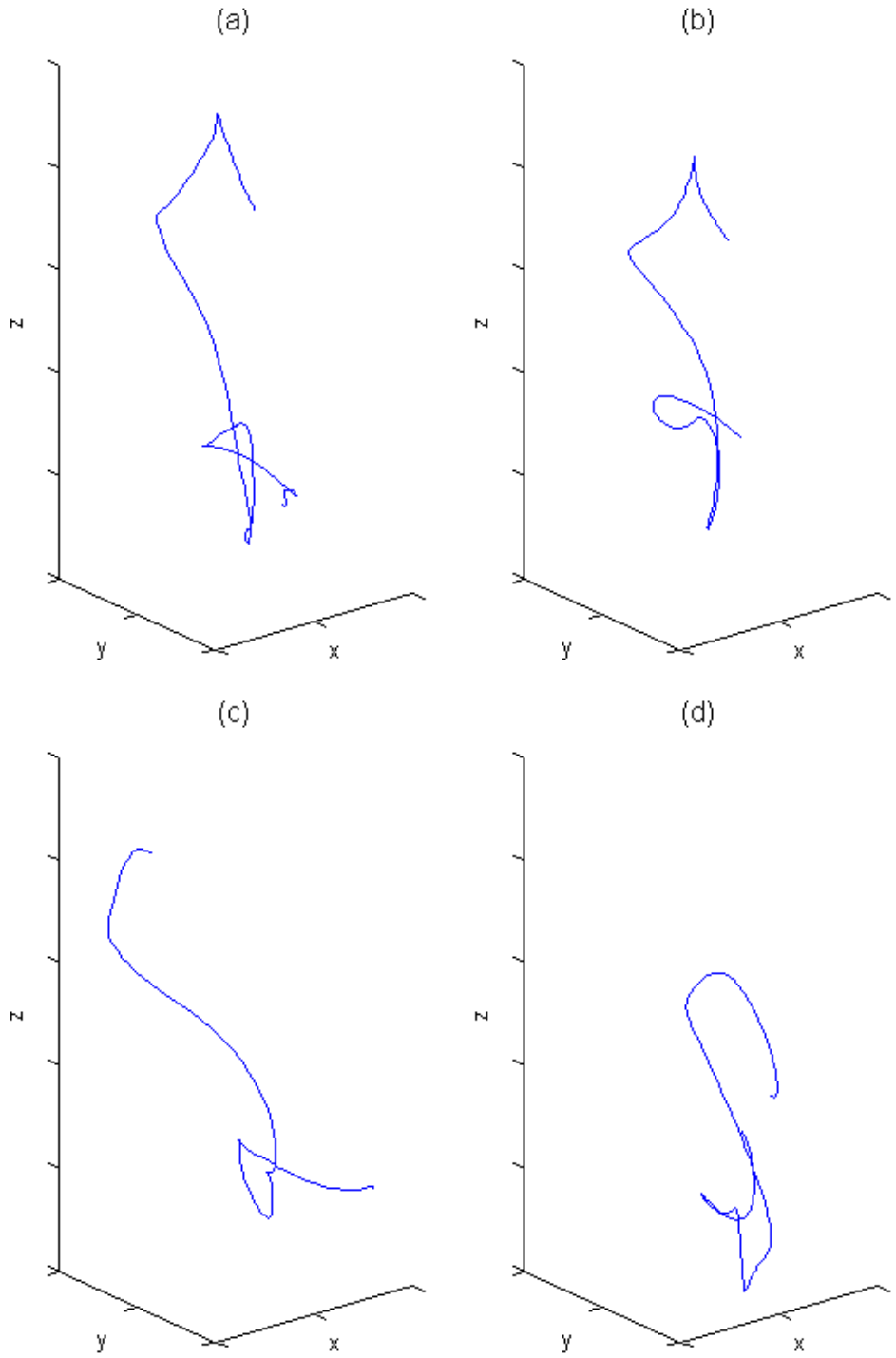


Fig. 5.9 – Signal $\boldsymbol{y}_{52}$: trajectoire originale (a) et approximations avec $K = 5$ atomes pour les cas non-orienté (b), orienté (c) et Mch-DCT (d).

en Fig. 5.11(b). En Fig. 5.11(a), nous observons que le noyau NOLD 6 est utilisé trois fois avec des orientations différentes (voir les angles d'Euler de la Fig. 5.8), alors que en Fig. 5.11(b), le noyau OLD 6 est utilisé deux fois, mais sans la possibilité de changer d'orientation pour mieux coïncider avec la trajectoire originale. Remarquons que dans le cas 3D orienté traité par M-OMP, un coefficient négatif $x_{l,\tau}$ donne une réflexion de la trajectoire associée.

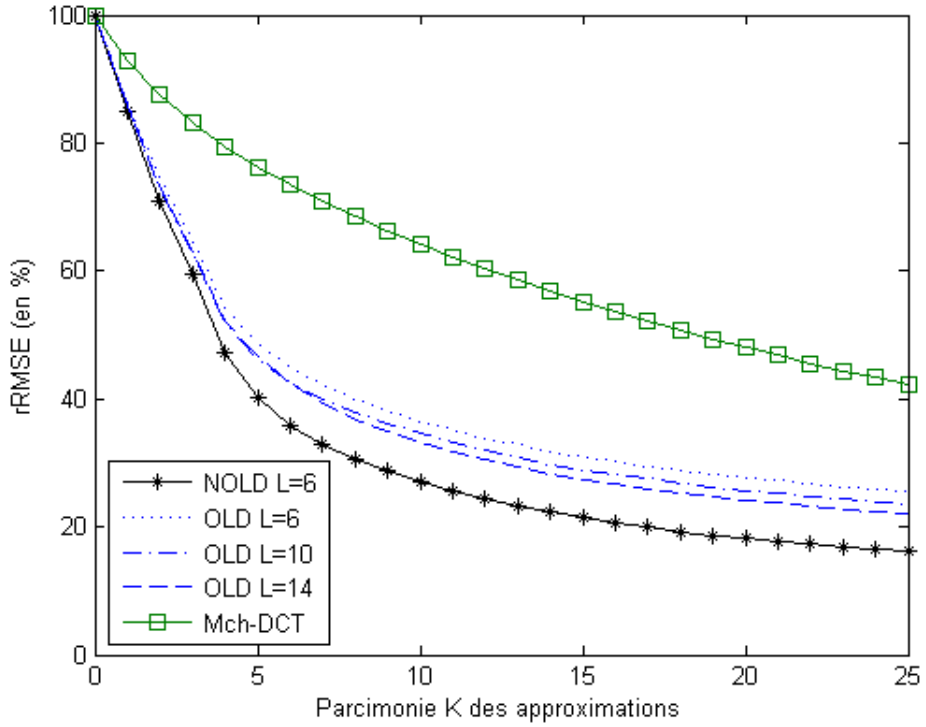


Fig. 5.10 – rRMSE sur les données LPC en fonction de la parcimonie K des approximations pour les différents dictionnaires.

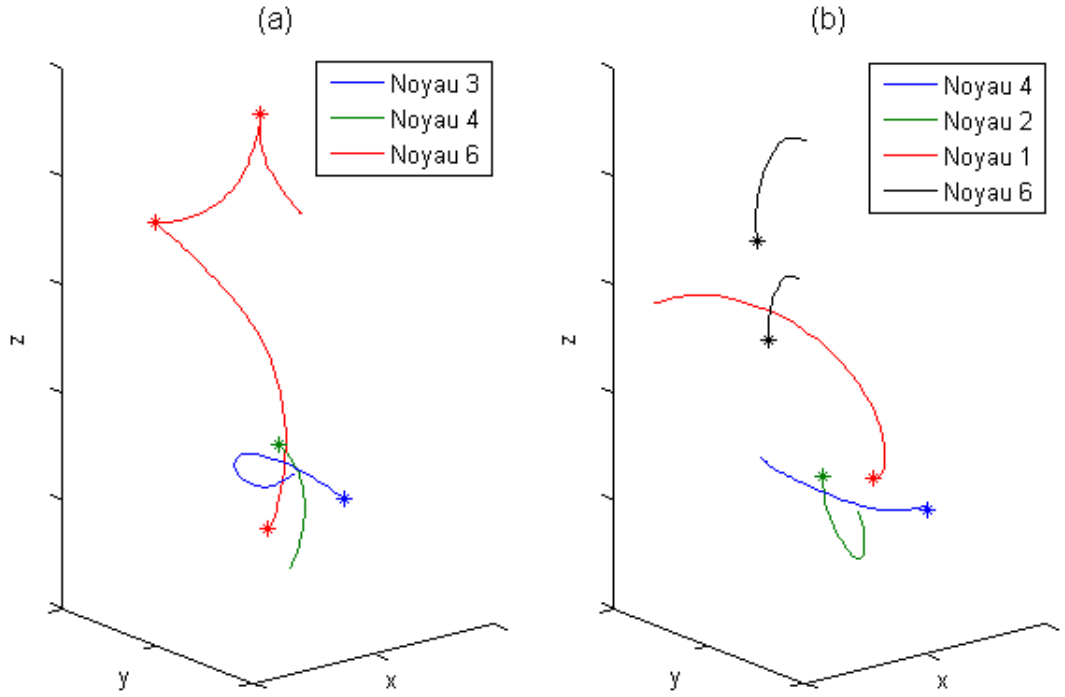


Fig. 5.11 – Signal $\boldsymbol{y}_{52}$: trajectoire reconstruite non-orientée (a) et trajectoire reconstruite orientée (b) en utilisant les atomes principaux.

Pour conclure cette expérience, nous avons observé que l'approche 3DRI fournit un dictionnaire de noyaux efficace et compact et favorise une meilleure coïncidence des noyaux en permettant leurs rotations.

5.8.4 Expérience 3 : décompositions des données tournées

Dans cette expérience, les signaux de la base de données sont tournés de différents angles aléatoires et ils sont notés $(.)'$. Les dictionnaires NOLD et OLDs appris dans l'expérience précédente sont gardés pour effectuer les décompositions sur les signaux tournés. Avec $K = 20$, la rRMSE moyennée sur les signaux est de 18.2% pour le cas non-orienté, de 48.5% pour le cas orienté avec $L = 6$, et 48.1% pour la Mch-DCT. Les rRMSE pour différentes valeurs de parcimonie K sont affichées en Fig. 5.12. En comparant les Fig. 5.10 et 5.12, nous remarquons que les courbes de rRMSE du NOLD sont identiques, ce qui prouve son invariance par rotation, contrairement aux OLDs dont les performances diminuent fortement. Nous observons aussi que le Mch-DCT est insensible aux rotations des données.

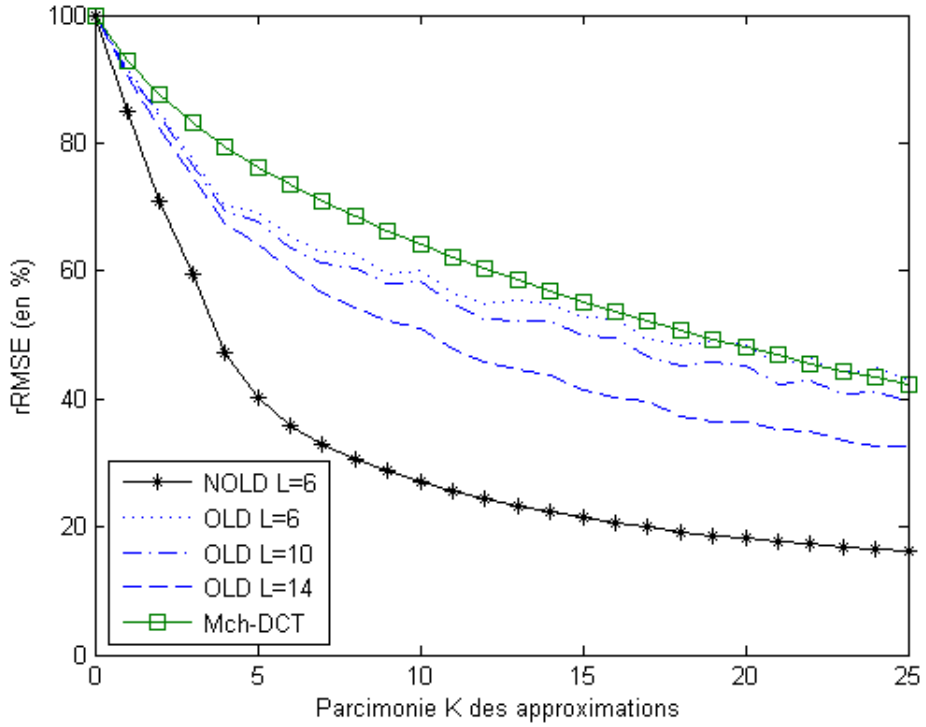


Fig. 5.12 – rRMSE sur les données LPC tournées en fonction de la parcimonie K des approximations pour les différents dictionnaires.

Le signal tourné $\boldsymbol{y}'_{52}$ est traité par le 3DRI-OMP avec le NOLD et est affiché en Fig. 5.13. Le signal tourné $\boldsymbol{y}'_{52}$ est représenté en (a), son approximation $\hat{\boldsymbol{y}}'_{52}$ en (b) avec $K = 10$ atomes, et le spikegramme associé en (c). Nous comparons maintenant les deux spikegrammes des Fig. 5.8(c) et Fig. 5.13(c) venant des décompositions non-orientées des signaux $\boldsymbol{y}_{52}$ et $\boldsymbol{y}'_{52}$. Les indices de noyaux, les paramètres de décalage et les amplitudes des coefficients sont les mêmes. Cependant, le mode de représentation visuelle des rotations ne permet pas d'identifier la rotation appliquée. Il n'est donc pas possible de voir si les différences d'angles correspondent à la rotation aléatoire appliquée au signal. La matrice de rotation aléatoire est notée R^r, la matrice de rotation estimée dans l'Expérience 2 est notée R^{e2} et R^{e3} pour

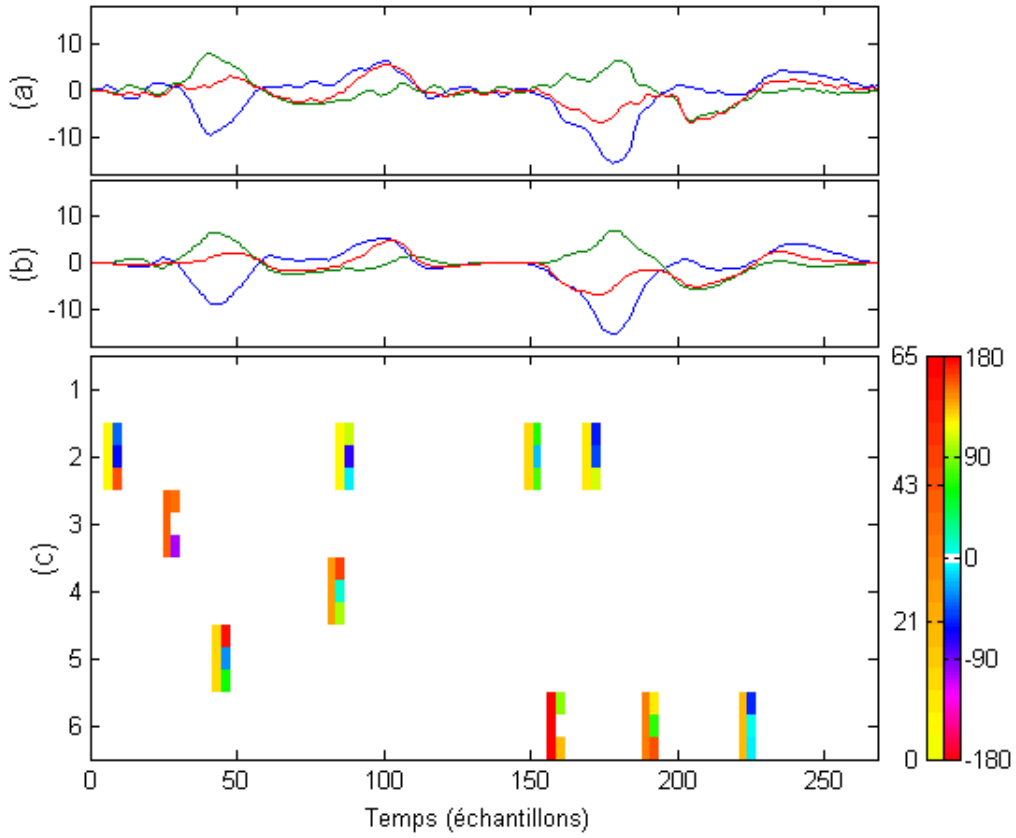


Fig. 5.13 – Signal de vitesse tourné $\boldsymbol{y}'_{52}$ (a) et son approximation $\hat{\boldsymbol{y}}'_{52}$ (b) et le spikegramme associé (c).

l'Expérience 3. Pour chaque signal $\boldsymbol{y}_q$, l'erreur e_q entre les matrices de rotation est calculée telle que :

$$e_q \;=\; \frac{1}{K}\sum_{k=1}^{K}\left\| R^{e3}_{l^k,\tau^k;\,q} - R^r_q\, R^{e2}_{l^k,\tau^k;\,q}\right\|^2_F . \tag{5.27}$$

Cette erreur est moyennée sur les Q signaux et elle est nulle. Ainsi, les différences entre les paramètres de rotation des Expériences 2 et 3 correspondent exactement aux rotations aléatoires appliquées aux signaux. Cela montre l'invariance par rotation 3D de la méthode non-orientée.

Comme dans l'expérience précédente, la trajectoire du signal tourné $\boldsymbol{y}'_{52}$ est reconstruite avec ses atomes principaux en Fig. 5.14. La trajectoire 3D de $\boldsymbol{y}'_{52}$ est tracée en Fig. 5.14(a), celle reconstruite non-orientée en Fig. 5.14(b), et celle reconstruite orientée en Fig. 5.14(c). Les reconstructions des Fig. 5.11(a) et Fig. 5.14(b) sont similaires en prenant en compte la rotation. Les reconstructions des Fig. 5.11(b) et Fig. 5.14(c) sont totalement différentes, notamment concernant les atomes utilisés. Cela montre la limitation d'un dictionnaire orienté fixé dans une orientation particulière, et le fait qu'il ne soit pas approprié pour des données aux orientations multiples.

L'avantage de l'approche non-orientée sur l'approche orientée a été montré dans la première expérience, avec de meilleurs résultats même sans rotation dans les données, et ces résultats sont encore plus nets dans la dernière expérience où les données sont tournées. L'approche non-orientée est robuste à la rotation : la rRMSE est identique et les atomes sélectionnés sont similaires quelle que soit la rotation. Pour conclure cette section, nos méthodes 3DRI ont été appliquées à l'analyse de trajectoires 3D, et des

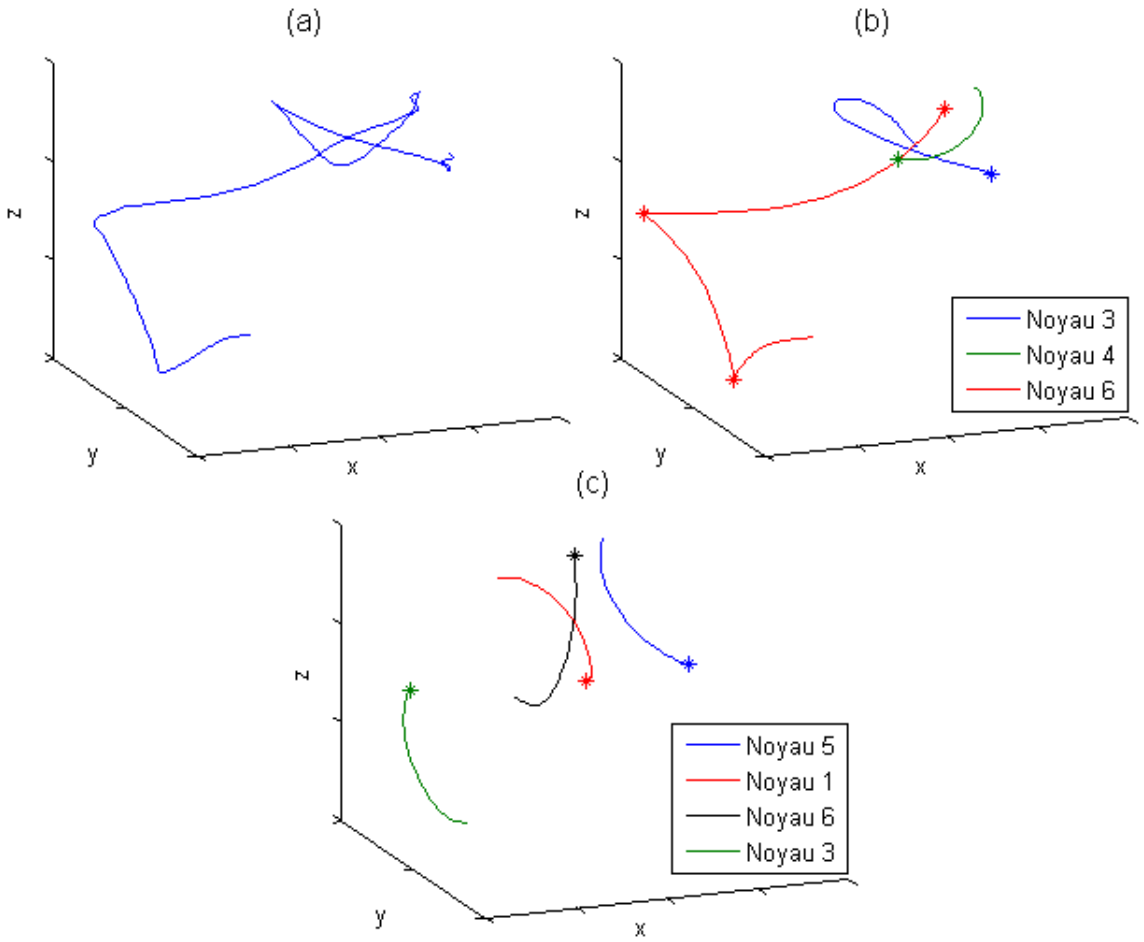


Fig. 5.14 – Signal tourné $\boldsymbol{y}'_{52}$: trajectoire originale (a), trajectoire reconstruite non-orientée (b), et trajectoire reconstruite orientée (c) en utilisant les atomes principaux.

expériences comparatives ont montré la pertinence de ces méthodes.

5.9 Extensions

Dans cette section, nous proposons deux extensions des méthodes 3DRI présentées : une extension pour des signaux multivariés avec $V > 3$, et une extension pour des signaux issus de P capteurs de mouvements 3D soumis à la même rotation.

5.9.1 Extension nDRI

Les méthodes présentées dans ce chapitre peuvent être facilement étendues en dimension supérieure $V > 3$, en considérant le signal multivarié $\boldsymbol{y} \in \mathbb{R}^{V \times N}$. Dans ce cas, la signification physique de la matrice orthogonale $R \in \mathbb{R}^{V \times V}$ est perdue.

En considérant les signaux multivariés, nous regardons quelles sont les étapes des algorithmes introduits qui sont modifiées. L'extension, que nous appelons nDRI-MP, modifie seulement le recalage 3D (étapes 4-5 du 3DRI-MP), en étendant la définition de la variable interne $\Sigma_2 = \mathrm{diag}(1, \ldots, 1, \det(UV^T)) \in \mathbb{R}^{V \times V}$. La procédure d'optimisation décrite en Section 5.4.2 est inchangée pour donner le nDRI-OMP.

Concernant l'apprentissage, l'Eq. (5.24) est inchangée pour donner la mise à jour du dictionnaire du nDRI-DLA, en utilisant le nDRI-OMP pour l'approximation parcimonieuse.

L'expérience réalisée en Section 5.7.2 est reprise intégralement telle quelle, mais en étendant les signaux au cas nDRI avec $V = 10$. Les taux de détection obtenus pour un seuil de 0.99 (nous rappelons que ce seuil a été proposé dans l'expérience standard de Aharon [Aha06, Chap. 6]) et moyennés sur 10 apprentissages sont résumés dans la Table 5.2. Nous rappelons que nDRI-DLA (a) donne les résultats pour l'ensemble d'apprentissage où les signaux sont tournés, nDRI-DLA (b) donne les résultats pour l'ensemble d'apprentissage où les signaux ne sont pas tournés, et M-DLA donne les résultats pour l'ensemble d'apprentissage où les signaux ne sont pas tournés. Nous observons que les taux de détection du nDRI-DLA sont quasi nuls, alors que ceux du M-DLA sont du même ordre que pour l'expérience avec $V = 3$ (cf. Table 5.1). L'augmentation du nombre de composantes V semble donc plus affecter le nDRI-DLA que le M-DLA.

TABLE 5.2 – Taux de détection (en %) pour un seuil de 0.99.

Niveau de bruit (en dB)	10	20	30	Pas de bruit
nDRI-DLA (a)	0	1.6	0.7	0.7
nDRI-DLA (b)	0	0	0	0
M-DLA	66.2	66.4	64.7	67.1

Nous abaissons alors le seuil de détection à 0.97 (ce qui reste un seuil exigent), et les résultats obtenus sont résumés en Table 5.3. Nous observons que les taux de détection pour le nDRI-DLA (a) sont bien meilleurs. Cela signifie que la plupart des noyaux sont quasi appris, mais que le seuil de 0.99 est trop élevé pour les détecter. La nature binaire du seuil de détection empêche une bonne visualisation de la qualité du dictionnaire appris. Cela provient du fait que ce taux de détection n'est pas une métrique au sens mathématique. Pour palier à ce problème, des métriques adaptées aux dictionnaires sont proposées dans [CBA13].

TABLE 5.3 – Taux de détection (en %) pour un seuil de 0.97.

Niveau de bruit (en dB)	10	20	30	Pas de bruit
nDRI-DLA (a)	56.2	87.6	82.7	84.0
nDRI-DLA (b)	0	0	0	0
M-DLA	78.9	78.2	76.9	78.9

5.9.2 Extension multicapteurs

Comme en Section 4.2.1, nous considérons P capteurs faisant l'acquisition de données trivariées : accéléromètre, gyromètre, etc. Ces capteurs sont physiquement liés et sont donc soumis à la même rotation.

Nous détaillons comment les algorithmes précédents sont étendus au cas multicapteurs. Nous définissons deux signaux suivants : $\boldsymbol{y} \in \mathbb{R}^{3\times N\mathsf{P}}$ obtenu par concaténation verticale des P signaux trivariés,

et $\boldsymbol{y}\rfloor \in \mathbb{R}^{3\mathsf{P}\times N}$ obtenu par concaténation horizontale des P signaux trivariés. Remarquons que l'un n'est pas le transposé de l'autre. Cette notation est aussi valable pour le résidu et les noyaux. Nous définissons aussi la matrice $R_{l,\tau}\rfloor \in \mathbb{R}^{3\mathsf{P}\times 3\mathsf{P}}$ telle que :

$$R_{l,\tau}\rfloor = \begin{bmatrix} R_{l,\tau} & & \\ & \ddots & \\ & & R_{l,\tau} \end{bmatrix} . \tag{5.28}$$

Le 3DRI-MP et le 3DRI-OMP multicapteurs sont obtenus en utilisant le signal $\boldsymbol{y}$, les noyaux $\{\boldsymbol{\psi}_l\}_{l=1}^{L}$ et les matrices $R_{l,\tau}$ dans les équations. Les rotations communes sont choisies conjointement entre les P signaux et l'approximation K-parcimonieuse obtenue peut se réécrire :

$$\hat{\boldsymbol{y}}^K\rfloor = \sum_{k=1}^{K} x^k_{l^k,\tau^k}\, R^k_{l^k,\tau^k}\rfloor\, \boldsymbol{\psi}_{l^k}(t-\tau^k)\rfloor \, . \tag{5.29}$$

Pour l'étape de mise à jour du dictionnaire du 3DRI-DLA, l'Eq. (5.24) est effectuée avec les noyaux $\{\boldsymbol{\psi}_l\rfloor\}_{l=1}^{L}$, les matrices $R_{l,\tau}\rfloor$, et le résidu $\boldsymbol{\epsilon}\rfloor$.

Si les P capteurs ne sont pas soumis à la même rotation, les signaux doivent être traités par un autre modèle, comme il sera évoqué dans les Perspectives.

Conclusion

Ce chapitre propose un nouveau modèle pour décrire des trajectoires 3D comme la somme de noyaux invariants par translation et par rotation 3D. Les 3DRI-MP et 3DRI-OMP permettent d'effectuer des décompositions invariantes à la rotation 3D en estimant les coefficients et les matrices de rotation. Ces méthodes ne sont pas l'extension triviale du cas 2D. Le 3DRI-DLA permet d'apprendre un dictionnaire de noyaux 3DRI. Une telle approche fournit un dictionnaire compact de noyaux capables de tourner, elle est robuste à la rotation et elle est plus efficace que l'état de l'art, même quand les signaux ne sont pas tournés. Après validation sur données simulées, ces algorithmes ont été appliqués sur des signaux de mouvement de Langage Parlé Complété. Cette approche a aussi été généralisée au cas nD.

Il existe plusieurs domaines d'application : *structure-from-motion* non-rigide, suivi 3D, représentation et analyse de gestes, apprentissage de primitives du mouvement pour des tâches spécifiques, et tout autre traitement basé sur un modèle 3DRI. Par exemple, en robotique, différentes études ont appris des primitives du mouvement pour des modes de locomotion spécifiques de robots bipèdes [HYK+01, TCA11, PZA12]. Cependant, un lourd formalisme est utilisé pour paramétrer la cinématique des robots. Il est possible de résoudre ce problème en utilisant l'apprentissage de dictionnaire qui est non-paramétrique et qui fait émerger les primitives du mouvement d'une base de données.

Bibliographie

[Aha06] M. Aharon : *Overcomplete Dictionaries for Sparse Representation of Signals.* Thèse de doctorat, Technion - Israel Institute of Technology, 2006.

[AHB87] K.S. Arun, T.S. Huang et S.D. Blostein : Least-squares fitting of two 3-D point sets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-9:698–700, 1987.

[ASKK11] I. Akhter, Y. Sheikh, S. Khan et T. Kanade : Trajectory space : A dual representation for nonrigid structure from motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33:1442–1456, 2011.

[BHB00] C. Bregler, A. Hertzmann et H. Biermann : Recovering non-rigid 3D shape from image streams. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR*, pages 690–696, 2000.

[BLM12] Q. Barthélemy, A. Larue et J.I. Mars : 3D rotation invariant decomposition of motion signals. *In Computer Vision – ECCV 2012*, volume 7585 de *Lecture Notes in Computer Science*, pages 172–182, 2012.

[BM92] P.J. Besl et H.D. McKay : A method for registration of 3-D shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14:239–256, 1992.

[BMB+09] J.P. Bandera, R. Marfil, A. Bandera, J.A. Rodriguez, L. Molina-Tanco et F. Sandoval : Fast gesture recognition based on a two-level representation. *Pattern Recognition Letters*, 30:1181–1189, 2009.

[BS98] R. Benjemaa et F. Schmitt : A solution for the registration of multiple 3D point sets using unit quaternions. *In Proc. Eur. Conf. Computer Vision ECCV '98*, pages 34–50, 1998.

[BSGL96] R. Bergevin, M. Soucy, H. Gagnon et D. Laurendeau : Towards a general multi-view registration technique. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:540–547, 1996.

[BSSS86] C.M. Bastuscheck, E. Schonberg, J.T. Schwartz et M. Sharir : Object recognition by 3-Dimensional curve matching. *Int. Journal of Intelligent Systems*, 1:105–132, 1986.

[CAS05] A. Croitoru, P. Agouris et A. Stefanidis : 3D trajectory matching by pose normalization. *In Proc. ACM Int. Workshop on Geographic Information Systems GIS '05*, pages 153–162, 2005.

[CBA13] S. Chevallier, Q. Barthélemy et J. Atif : Metrics for multivariate dictionaries. Preprint arXiv :1302.4242, 2013.

[ELF97] D.W. Eggert, A. Lorusso et R.B. Fisher : Estimating 3-D rigid body transformations : a comparison of four major algorithms. *Machine Vision and Applications*, 9:272–290, 1997.

[GBB+05] G. Gibert, G. Bailly, D. Beautemps, F. Elisei et R. Brun : Analysis and synthesis of the three-dimensional movements of the head, face, and hand of a speaker using cued speech. *Journal of the Acoustical Society of America*, 118:1144–1153, 2005.

[GD04] J.C. Gower et G.B. Dijksterhuis : *Procrustes Problems*. Oxford Statistical Science Series, 30, 2004.

[Gib06] G. Gibert : *Conception et évaluation d'un système de synthèse 3D de Langue française Parlée Complétée (LPC) à partir du texte*. Thèse de doctorat, Institut National Polytechnique de Grenoble, 2006.

[Gow75] J.C. Gower : Generalized Procrustes analysis. *Psychometrika*, 40:33–51, 1975.

[Gow10] J.C. Gower : Procrustes methods. *Wiley Interdisciplinary Reviews : Computational Statistics*, 2:503–508, 2010.

[GRSV07] R. Gribonval, H. Rauhut, K. Schnass et P. Vandergheynst : Atoms of all channels, unite ! Average case analysis of multi-channel sparse recovery using greedy algorithms. Rapport technique PI-1848, IRISA, 2007.

[Han06] A.J. Hanson : *Visualizing quaternions*. Morgan-Kaufmann/Elsevier, 2006.

[Hor87] B.K.P. Horn : Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4:629–642, 1987.

[Hor88] B.K.P. Horn : Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 4:1127–1135, 1988.

[HYK+01] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi et K. Tanie : Planning walking patterns for a biped robot. *IEEE Trans. on Robotics and Automation*, 17:280–289, 2001.

[Kan94] K. Kanatani : Analysis of 3-D rotation fitting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16:543–549, 1994.

[KPJR91] B. Kamgar-Parsi, J.L. Jones et A. Rosenfeld : Registration of multiple overlapping range images : scenes without distinctive features. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:857–871, 1991.

[NA05] Y. Nishimori et S. Akaho : Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing*, 67:106–135, 2005.

[Plu05] M.D. Plumbley : Geometrical methods for non-negative ICA : Manifolds, Lie groups and toral subalgebras. *Neurocomputing*, 67:161–197, 2005.

[PP08] K.B. Petersen et M.S. Pedersen : The matrix cookbook. Rapport technique, Technical University of Denmark, 2008.

[PZA12] M.J. Powell, H. Zhao et A.D. Ames : Motion primitives for human-inspired bipedal robotic locomotion walking and stair climbing. *In Proc. IEEE Int. Conf. Robotics and Automation ICRA*, pages 543–549, 2012.

[RL01] S. Rusinkiewicz et M. Levoy : Efficient variants of the ICP algorithm. *In Proc. Int. Conf. on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

[Sch66] P.H. Schonemann : A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31:1–10, 1966.

[SLY10] Y. Su, L. Liu et Y. Yang : Optimal trajectory space finding for nonrigid structure from motion. *In Advanced Concepts for Intelligent Vision Systems ACIVS '10*, volume 6474 de *Lecture Notes in Computer Science*, pages 357–366, 2010.

[SS85] J.T. Schwartz et M. Sharir : Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. Rapport technique Robotics Report 46, Courant Institute of Mathematical Sciences, New York University, 1985.

[TCA11] D. Tlalolini, C. Chevallereau et Y. Aoustin : Human-like walking : Optimal motion of a bipedal robot with toe-rotation motion. *IEEE/ASME Trans. on Mechatronics*, 16:310–320, 2011.

[THB04] L. Torresani, A. Hertzmann et C. Bregler : Learning non-rigid 3D shape from 2D motion. *In Advances in Neural Information Processing Systems NIPS '04*, pages 1555–1562, 2004.

[THB08] L. Torresani, A. Hertzmann et C. Bregler : Nonrigid structure-from-motion : Estimating shape and motion with hierarchical priors. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30:878–892, 2008.

[Ume91] S. Umeyama : Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:376–380, 1991.

[WH60] B. Widrow et M.E. Hoff : Adaptive switching circuits. *In Proc. WESCON Conv. Rec.*, pages 96–104, 1960.

[Wid71] B. Widrow : *Adaptive Filters*, pages 563–586. Aspects of Network and System Theory, 1971.

[WSV91] M.W. Walker, L. Shao et R.A. Volz : Estimating 3-D location parameters using dual number quaternions. *CVGIP : Image Understanding*, 53:358–367, 1991.

[ZSD12] M. Zhu, H. Sun et Z. Deng : Quaternion space sparse decomposition for motion compression and retrieval. *In Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '12, pages 183–192, 2012.

[ZZZ+12] S. Zhang, Y. Zhan, Y. Zhou, M.G. Uzunbas et D.N. Metaxas : Shape prior modeling using sparse representation and online dictionary learning. *In Medical Image Computing and Computer-Assisted Intervention - MICCAI 2012*, volume 7512 de *Lecture Notes in Computer Science*, pages 435–442, 2012.

Chapitre 6

Classification invariante par translation

Introduction

Dans les chapitres précédents, nous nous sommes focalisés sur une approche générative/reconstructive des données. Mais, dans la grande majorité des cas, l'objectif final reste la classification des signaux qui correspond à une approche discriminative des données. Ce chapitre fait le lien entre ces deux domaines.

Dans des travaux récents [Mal12, BM13], Mallat insiste sur le fait que la clé pour faire de la classification efficacement n'est pas la parcimonie des représentations, mais leurs invariances. Dans cette thèse, nous avons présenté des algorithmes fournissant des représentations invariantes à la translation temporelle (paramètre τ), à l'échelle/dilatation [1] (paramètre $|x_{l,\tau}|$), à la rotation (angle $\theta_{l,\tau}$ en 2D ou matrice de rotation $R_{l,\tau}$ en 3D), et à la translation spatiale (utilisation des signaux de vitesse au lieu des signaux de position) pour des signaux multivariés. D'autre part, comme déjà évoqué dans les deux chapitres précédents, les méthodes usuelles de reconnaissance de trajectoire utilisent des descripteurs invariants par rotation [CAS05, BMB+09] perdant ainsi l'information de la rotation contrairement à nos algorithmes de décomposition qui estiment les paramètres de rotation. Forts de ces considérations prometteuses, nous voulons ajouter une étape de classification qui puisse exploiter les représentations issues de nos algorithmes de décomposition : M-OMP, 2DRI-OMP et 3DRI-OMP.

Dans ce chapitre, l'invariance par rotation n'est pas traitée. Nous nous intéressons donc à tous les signaux temporels qui doivent être classés avec la contrainte d'invariance par translation. Dans un premier temps, nous étudierons les descripteurs consistants par translation, utiles pour traiter des décompositions temporelles. Nous comparerons ensuite ces différents descripteurs entre eux avec le même classifieur et sur les mêmes données. Nous introduirons enfin l'apprentissage de dictionnaire discriminant, en faisant des liens avec les travaux existants dans le cas de l'invariance par translation.

6.1 Descripteurs consistants par translation

Dans cette section, nous définissons le problème d'extraction de descripteurs consistants par translation, puis nous ferons l'état de l'art des fonctions répondant à cette problématique.

1. Bien noter qu'il s'agit ici d'une dilatation spatiale et non temporelle.

6.1.1 Formulation du problème

Nous voulons classer les signaux $\boldsymbol{Y} = \{\boldsymbol{y}_p\}_{p=1}^{P}$ en C classes, notées Cl_c avec $c \in \mathbb{N}_C$. De plus, nous cherchons à intégrer l'invariance par translation de la représentation dans cette étape de classification. C'est-à-dire, nous souhaitons que deux signaux identiques, dont l'un est seulement la version translatée de l'autre, soient rangés dans la même classe. Nous rappelons le modèle de la décomposition K-parcimonieuse invariante par translation :

$$\boldsymbol{y}_p(t) = \sum_{k=1}^{K} x_{l_p^k,\tau_p^k}\, \boldsymbol{\psi}_{l_p^k}(t-\tau_p^k) + \boldsymbol{\epsilon}(t)\,. \tag{6.1}$$

Ce modèle est invariant par dilatation grâce aux coefficients $x_{l_p^k,\tau_p^k}$ et par translation temporelle grâce aux décalages τ_p^k des noyaux donnant les atomes $\boldsymbol{\psi}_{l_p^k}(t-\tau_p^k)$. Nous souhaitons ajouter une étape de classification adaptée à de telles invariances.

Nous considérons le spikegramme du signal $\boldsymbol{y}_p$ composé de K coefficients actifs : $\left\{x_{l_p^k,\tau_p^k}\right\}_{k=1}^{K}$. Nous cherchons donc à extraire un vecteur de caractéristiques v_p (aussi appelé descripteur) de ce spikegramme. Nous souhaitons que le classifieur, qui traitera ensuite tous les vecteurs $\{\mathsf{v}_p\}_{p=1}^{P}$, effectue une classification robuste à la translation. Ainsi, pour extraire ces caractéristiques, nous cherchons une fonction consistante par translation. Nous rappelons les deux définitions complémentaires introduites en Section 1.1.3 :

- invariance par translation : si $y(t) \mapsto x(t)$, alors $y(t+t_0) \mapsto x(t+t_0)$,
- consistance par translation : si $y(t) \mapsto x(t)$, alors $y(t+t_0) \mapsto x(t)$.

Dans notre cas, la décomposition des signaux est invariante par translation, et la robustesse à la translation du classifieur est obtenue par la consistance par translation des descripteurs :

$$\text{si} \quad \left\{x_{l_p^k,\tau_p^k}\right\}_{k=1}^{K} \mapsto \mathsf{v}_p, \quad \text{alors} \quad \left\{x_{l_p^k,\tau_p^k+\tau_0}\right\}_{k=1}^{K} \mapsto \mathsf{v}_p\,. \tag{6.2}$$

Pour résumer, une classification invariante par translation s'obtient par l'emploi d'un classifieur sur des descripteurs consistants par translation, extraits de représentations invariantes par translation. C'est la succession de l'invariance et de la consistance qui assure une classification robuste à la translation.

Nous allons passer en revue les différentes fonctions permettant d'intégrer temporellement l'information d'un spikegramme, puis nous comparerons ces différentes fonctions. Nous illustrerons cette démarche sur la base de données UCI *Character Trajectories* déjà utilisée pour illustrer l'approche 2DRI (cf. Section 4.3.1).

6.1.2 Etat de l'art sur les fonctions de groupement

Nous cherchons des fonctions qui extraient d'un spikegramme un vecteur de caractéristiques répondant aux critères évoqués ci-dessus. Une fonction de groupement, *pooling function* en anglais, est une fonction d'intégration qui calcule un vecteur de caractéristiques sur un voisinage, de telle sorte que la localisation exacte des éléments sur ce voisinage ne rentre pas en compte dans le calcul [LeC12]. Ainsi, seule l'appartenance au voisinage importe, et non la localisation exacte ou l'ordre. De plus, la fonction de groupement réduit la dimension des vecteurs étudiés, effectuant ainsi un sous-échantillonnage.

Considérant le signal $\boldsymbol{y}_p$, le vecteur de caractéristiques $\mathsf{v}_p \in \mathbb{R}^L$ est calculé pour chaque noyau l par

la fonction de groupement suivante :

$$\mathsf{v}_p(l) = \sum_{k=1}^{K} \left| x_{l_p^k, \tau_p^k} \right|^s \quad \text{t.q.} \quad l_p^k = l \,, \tag{6.3}$$

pour des valeurs de $s = 0,\ 0.5,\ 1,\ 2$ [GRKN07] et $+\infty$ (équivalente au *max-pooling*) [HYHJ+12]. Cette fonction est équivalente à une norme ℓ_s : $\mathsf{v}_p(l) = \|x_{l,.}\|_s^s$. Remarquons que dans [SP11], le vecteur de caractéristiques final est la concaténation des vecteurs obtenus avec $s = 1$ et $s = 0$. Toutefois, les fonctions décrites en Eq. (6.3) engendrent une perte d'information, notamment sur l'ordre temporel des coefficients. Grosse *et al.* suggèrent l'idée d'utiliser les fonctions de groupement sur des fenêtres décalées [GRKN07], créant ainsi plusieurs voisinages au lieu d'un seul, mais aucun détail d'implémentation n'est donné.

Dans [MBOL12], Mayoue *et al.* décrivent comment appliquer une fonction de groupement sur F fenêtres temporelles de Hanning, indicées par $f \in \mathbb{N}_F$. Une fenêtre de Hanning de longueur temporelle T_H est définie par :

$$\text{Hanning}(t) = \begin{cases} \frac{1}{2}\left(1 + \cos\left(\frac{2\pi}{T_H}\, t\right)\right), & \text{pour } t \in \left[-\frac{T_H}{2}, \frac{T_H}{2}\right] \\ 0 & \text{sinon} \end{cases} . \tag{6.4}$$

La taille T_H est calculée à partir du signal le plus long. Le temps maximum de la base de données $\boldsymbol{Y}$ est calculé tel que :

$$N_{max} = \max_p \left\{ \max_k \left\{ \tau_p^k \right\}_{k=1}^{K} \right\}_{p=1}^{P} . \tag{6.5}$$

Les F fenêtres sont disposées avec un recouvrement de 50 %, soit de $T_H \times 0.5$. Ainsi, leur taille est égale à $T_H = \frac{N_{max}}{0.5 \times F}$. L'inconvénient de l'approche fenêtrée est qu'il faut choisir la position temporelle de la première fenêtre. Ici, le centre de la première fenêtre est placé sur le temps $\tilde{\tau}$ d'apparition du premier atome de chaque spikegramme : $\tilde{\tau}_p = \min_k \left\{ \tau_p^k \right\}_{k=1}^{K}$. L'information du spikegramme est intégrée sur une matrice $\mathsf{V}_p \in \mathbb{R}^{L \times F}$ définie telle que :

$$\mathsf{V}_p(l, f) = \sum_{k=1}^{K} \text{Hanning}\left(\tau_p^k - (f-1) \times \frac{T_H}{2} - \tilde{\tau}_p\right) \times x_{l_p^k, \tau_p^k} \quad \text{t.q.} \quad l_p^k = l \,. \tag{6.6}$$

Ainsi, chaque élément $\mathsf{V}_p(l, f)$ correspond à la projection des atomes associés au noyau l sur la fenêtre de Hanning centrée en $(f-1) \times \frac{T_H}{2} + \tilde{\tau}_p$. La fonction de groupement qui effectue la somme est aussi appelée *mean-pooling* en anglais. A la fin, chaque matrice V_p est vectorisée dans un vecteur $\mathsf{v}_p \in \mathbb{R}^{1 \times LF}$:

$$\mathsf{v}_p((l-1) \times F + f) \leftarrow \mathsf{V}_p(l, f) \,. \tag{6.7}$$

La consistance par dilatation s'obtient facilement en normalisant chaque vecteur v_p obtenu :

$$\mathsf{v}_p \leftarrow \frac{\mathsf{v}_p}{\|\mathsf{v}_p\|_2} \,. \tag{6.8}$$

Ces vecteurs de caractéristiques normalisés peuvent être maintenant utilisés comme entrées des systèmes d'apprentissage automatique [HTF09], comme les SVM (*Support Vector Machine*), MLP (*Multi-Layer*

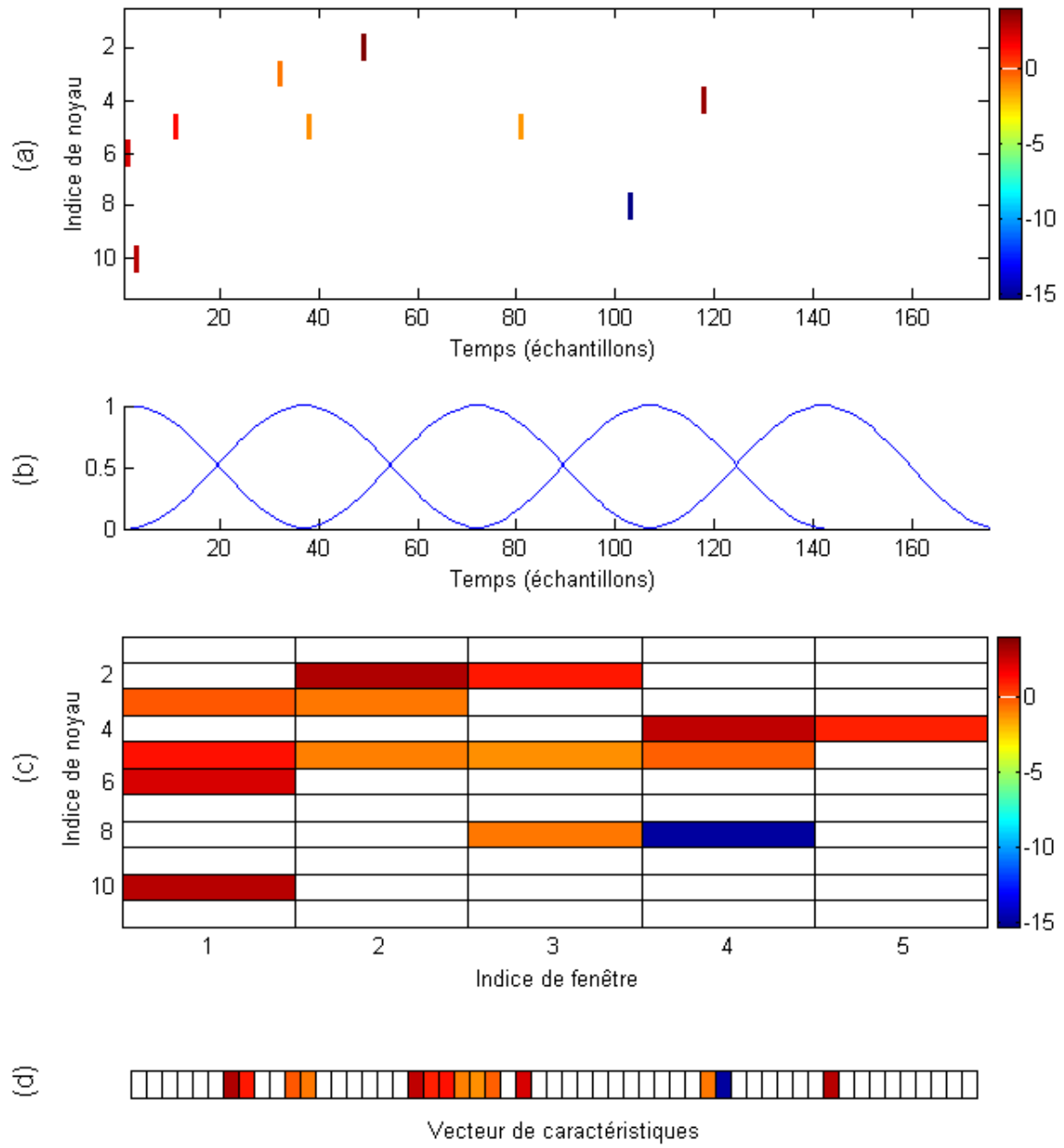


Fig. 6.1 – Illustration du traitement d'un spikegramme : le spikegramme initial du signal (a), les $F = 5$ fenêtres de Hanning (b), la matrice V_p obtenue par projection du spikegramme sur les fenêtres de Hanning (c) et le vecteur de caractéristiques v_p final (d).

Perceptron), etc. L'un des avantages de ce traitement est que la taille des vecteurs v_p est indépendante du nombre d'atomes K de la décomposition parcimonieuse.

L'ensemble de ce traitement est illustré en Fig. 6.1 pour une occurrence de la lettre a : le spikegramme initial du signal, obtenu par décomposition avec le M-OMP, est représenté en (a), les $F = 5$ fenêtres de Hanning en (b), la matrice V_p obtenue par projection du spikegramme sur les fenêtres de Hanning en (c) et le vecteur de caractéristiques v_p final en (d). Un exemple de vecteur de caractéristiques de chaque lettre/classe de la base de données est affiché en Fig. 6.2. Certains éléments sont communs à toutes les classes, d'autres sont plus spécifiques, et donc discrimants. Les 80 vecteurs de caractéritiques obtenus

pour les signaux de la lettre a sont maintenant illustrés Fig. 6.3. Au sein de cette même classe, nous constatons une bonne répétition des descripteurs, en particulier les colonnes 6, 19 et 26. Cela provient de la bonne reproductibilité des décompositions adaptées, constatée en Section 4.3.3 et 4.3.4, qui sont ensuite projetées sur les fenêtres de Hanning.

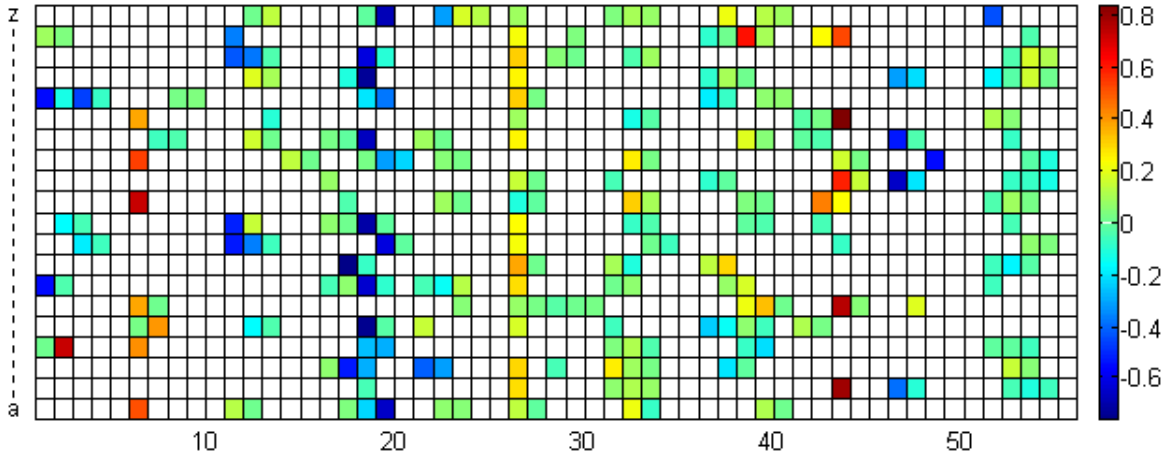


Fig. 6.2 – Illustration d'un vecteur de caractéristiques pour chaque classe, de la lettre a à la lettre z.

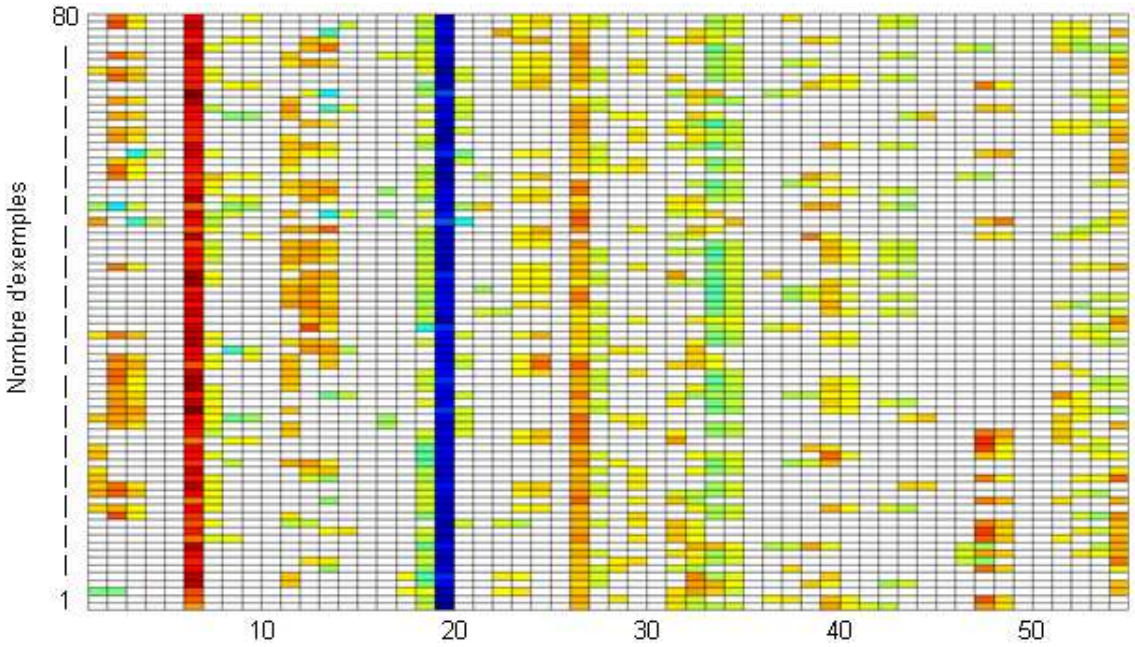


Fig. 6.3 – Illustration des 80 vecteurs de caractéristiques associés à tous les signaux représentant la lettre a : chaque ligne correspond à un vecteur v.

Par analogie, les fonctions de groupement décrites dans l'Eq. (6.3) sont appliquées sur les F fenêtres de Hanning, en utilisant l'approche précédemment décrite. Seule la définition de la matrice V_p change :

$$\mathsf{V}_p(l,f) = \sum_{k=1}^{K} \text{Hanning}\left(\tau_p^k - (f-1) \times \frac{T_H}{2} - \tilde{\tau}_p\right) \times \left|x_{l_p^k,\tau_p^k}\right|^s \quad \text{t.q.} \quad l_p^k = l\,. \tag{6.9}$$

Remarquons qu'il existe d'autres approches pour traiter les spikegrammes, notamment celles qui

ne font pas d'intégration temporelle des coefficients. Dans [MBM+12], Mouraud *et al.* exploitent des méthodes issues du domaine des neurosciences et décrivent une métrique entre deux trains d'impulsions multisources, *i.e.* des spikegrammes. Elle calcule la transformation minimale entre les deux spikegrammes en calculant les coûts de décalage temporel, de suppression et d'ajout de coefficients, ainsi que de changement de noyaux. L'inconvénient est que cette approche ne prend pas en compte l'amplitude des coefficients mais seulement leur signe.

Nous venons de faire la revue des méthodes d'extraction de caractéristiques consistantes par translation, mais elles n'ont jamais été comparées entre elles avec le même classifieur et les mêmes données. Grosse *et al.* utilisent des classifieurs SVM, GDA (*Gaussian Discriminant Analysis*) et MultiExp (*Multinomial Exponentials*) sur des signaux audio [GRKN07], Mayoue *et al.* utilisent des MLP sur des signaux mouvements [MBOL12], Huang *et al.* utilisent des SVM sur des signaux audio [HYHJ+12]. Nous souhaitons donc faire une comparaison de ces différentes fonctions.

6.2 Comparaisons des différentes fonctions de groupement

Dans cette section, nous comparons les différentes fonctions de groupement en utilisant le même classifieur (à savoir un SVM) sur les descripteurs extraits des mêmes décompositions parcimonieuses des signaux de la base de données UCI *Character Trajectories*. Nous testons aussi leurs consistances par dilatation et par translation.

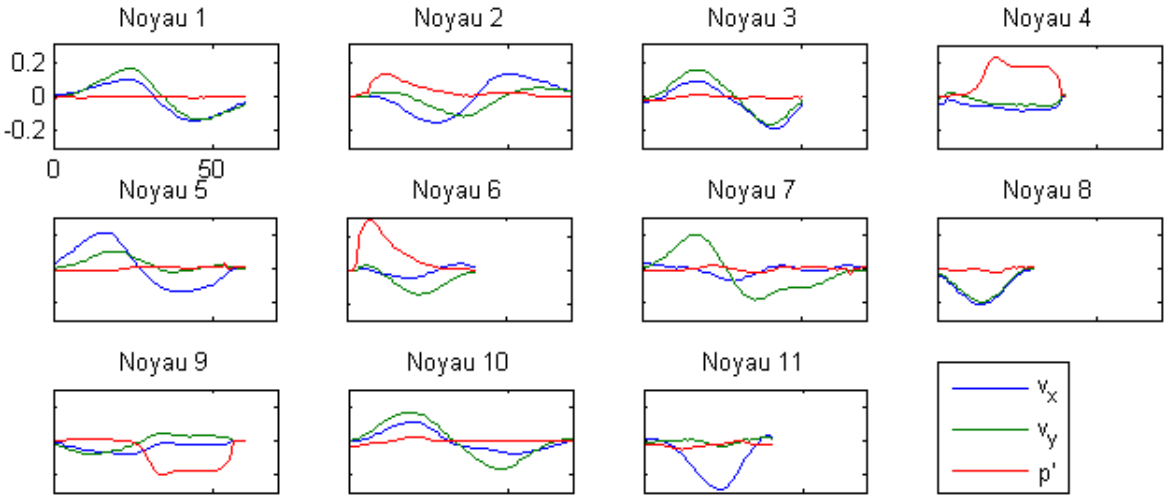


Fig. 6.4 – Dictionnaire de $L = 11$ noyaux appris par le M-DLA sur les signaux trivariés (vitesses cartésiennes et pression).

6.2.1 Apprentissages des classifieurs et comparaisons

Dans la suite de ce chapitre, les méthodes sont appliquées sur la base de données UCI *Character Trajectories* décrite en Section 4.3.1. Cette base est composée de $P = 1430$ signaux d'apprentissage $\boldsymbol{Y}_a$ et de $Q = 1425$ signaux de test $\boldsymbol{Y}_t$, repartis en $C = 20$ classes. Nous apprenons un dictionnaire de $L = 11$ noyaux multivariés (vitesses cartésiennes et pression) avec le M-DLA utilisé dans un cas trivarié

(cf. Chapitre 2). Le dictionnaire est illustré en Fig. 6.4. Nous calculons ensuite les décompositions des signaux, puis leurs vecteurs de caractéristiques pour les différentes fonctions de groupement précédemment étudiées. Les vecteurs de caractéristiques sont répartis entre l'ensemble d'apprentissage $\mathbf{V}_a$ et l'ensemble de test $\mathbf{V}_t$.

Algorithm 14 : η = Processus_N-*fold*_CV ($\mathbf{V}_a, \mathbf{V}_t$)

initialisation
for $\mathsf{c} \leftarrow 1, \mathsf{C}$ **do**
  for $\sigma \leftarrow 1, \Sigma$ **do**
    Apprentissage de C classifieurs multiclasses par N-*fold* CV sur $\mathbf{V}_a$:
    for $\mathsf{n} \leftarrow 1, \mathsf{N}$ **do**
      Définition des ensembles d'entraînement et de validation
      Apprentissage sur ensemble d'entraînement : *svmmulticlassoneagainstall* (c, σ)
      Evaluation sur l'ensemble de validation : $\eta_{validation} \leftarrow$ *svmmultival* (c, σ)
    end for
    $H(\mathsf{c}, \sigma) \leftarrow$ taux de classification $\bar{\eta}_{validation}$ moyenné sur les N essais
  end for
end for
Paramètres optimaux : $(\mathsf{c}_{opt}, \sigma_{opt}) \leftarrow \arg\max_{\mathsf{c},\sigma} H$
Apprentissage d'un SVM de paramètres c_{opt} et σ_{opt} sur l'ensemble de test $\mathbf{V}_t$
Taux de classification η obtenu sur $\mathbf{V}_t$

Nous apprenons un SVM multiclasse à noyaux Gaussiens [HTF09] en mode un contre tous sur l'ensemble d'apprentissage $\mathbf{V}_a$. Deux paramètres sont optimisés : le paramètre de coût ou compromis c et l'écart type du noyau σ. Nous faisons une validation croisée (CV) de type N-*fold* sur une grille de paramètres de $\mathsf{C} = 18$ valeurs de c et de $\Sigma = 15$ valeurs de σ, avec $\mathsf{N} = 5$. Le taux de classification η est calculé sur l'ensemble de test $\mathbf{V}_t$. Cet apprentissage est réalisé pour les différentes fonctions de groupement décrites précédemment, mais avec la même permutation aléatoire de l'ensemble d'apprentissage $\mathbf{V}_a$. Pour la validation croisée, $\mathbf{V}_a$ est divisé en un ensemble d'entraînement (de taille 4/5) et un ensemble de validation (de taille 1/5) qui tournent successivement. Ce processus, résumé dans l'Algorithme 14, est moyenné 10 fois et nous relevons le taux de classification moyen $\bar{\eta}$.

TABLE 6.1 – Taux de classification $\bar{\eta}$ sur les vecteurs de caractéristiques extraits des signaux.

Fonctions	ℓ_0	$\ell_{0.5}$	ℓ_1	ℓ_2	ℓ_∞	$\sum(.)$
Taux $\bar{\eta}$ (%)	38.37 ± 0.60	59.76 ± 0.74	65.00 ± 0.46	64.18 ± 0.19	66.42 ± 0.40	76.40 ± 0.94
Fonctions fenêtrées	ℓ_0	$\ell_{0.5}$	ℓ_1	ℓ_2	ℓ_∞	$\sum(.)$
Taux $\bar{\eta}$ (%)	60.19 ± 0.38	79.96 ± 0.13	82.53 ± 0.36	78.44 ± 0.83	83.54 ± 0.29	90.18 ± 0.21

La Table 6.1 résume les taux de classification moyens obtenus par les différentes fonctions de groupement. Les fonctions de groupement sont abrégées par leurs normes ℓ_s associées, et la fonction introduite par Mayoue *et al.* est notée $\sum(.)$. Nous observons que les fonctions issues de normes fenêtrées ont de meilleurs résultats que celles non fenêtrées. Les normes intègrent temporellement les coefficients sur tout

le spikegramme, perdant ainsi toute information liée à l'ordre temporel. Au contraire, les fenêtres préservent la localisation en temps des coefficients. L'avantage des fenêtres de Hanning décalées de moitié est qu'il n'y a pas de perte d'information tant qu'il n'y a pas plus de deux atomes issus du même noyau dans la même fenêtre. En effet, en considérant l'Eq. (6.9), nous avons :

$$\mathsf{V}_p(l,f) + \mathsf{V}_p(l,f+1) = \left|x_{l_p^k,\tau_p^k}\right|^s \quad \text{pour} \quad l_p^k = l \quad \text{et} \quad \tau_p^k \in \left[\tilde{\tau} + (f-1)\frac{T_H}{2}, \tilde{\tau} + f\frac{T_H}{2}\right] . \tag{6.10}$$

D'autre part, les fonctions de groupement issues de normes (fenêtrées ou pas) perdent le signe des coefficients, alors que ce n'est pas le cas de la fonction $\sum(.)$, d'où le meilleur taux de classification : 90.18 ± 0.21 %. Remarquons enfin que les variances des taux de classification sur les 10 expériences sont plutôt faibles.

Après avoir comparé les différentes fonctions de groupement, nous allons tester leurs consistances.

6.2.2 Tests de consistances

Nous souhaitons tester et illustrer les consistances par dilatation et par translation temporelle des fonctions de groupement. Dans un premier temps, nous testons la consistance par dilatation. Pour chaque signal de l'ensemble de test $\boldsymbol{Y}_t$, nous appliquons un coefficient multiplicatif tiré aléatoirement. Les vecteurs de caractéristiques $\mathbf{V}_t$ sont calculés, et le taux de classification est calculé avec les paramètres de SVM appris en Section 6.2.1. Nous n'affichons pas les résultats, car ils sont tous strictement similaires à la Table 6.1. Ces résultats prouvent effectivement la consistance par dilatation, obtenue par construction lors de la normalisation réalisée en Eq. (6.8).

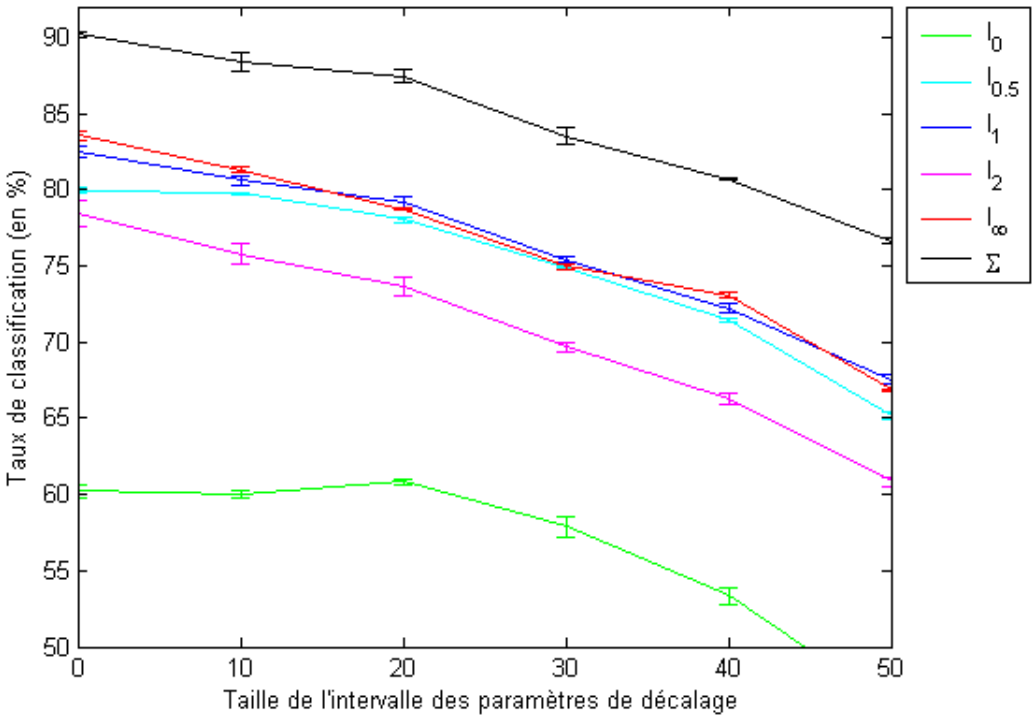


Fig. 6.5 – Taux de classification $\bar{\eta}_{consist}$ sur les vecteurs de caractéristiques extraits des signaux translatés aléatoirement pour les fonctions de groupement fenêtrées.

Dans un deuxième temps, nous testons si les fonctions de groupement génèrent bien la consistance par translation. Pour chaque signal de $\boldsymbol{Y}_t$, nous appliquons un décalage temporel (complétion de zéro) aléatoire, tiré uniformément sur un intervalle de taille $N_Z = 10$, 20, 30, 40 et 50. Les signaux étant de longueur moyenne $N = 250$, un décalage de 50 échantillons représente à peu près 1/5 du signal. Les vecteurs de caractéristiques $\mathbf{V}_t$ sont calculés, et le taux de classification $\eta_{consist}$ est calculé avec les paramètres de SVM appris en Section 6.2.1. Les résultats pour les fonctions de groupement non fenêtrées sont identiques à ceux de la Table 6.1 quelle que soit la valeur de N_Z, ce qui est dû au fait que le temps n'intervient pas Eq. (6.3). Nous rappelons que le taux de classification maximal est obtenu par la fonction $\sum(.)$ qui donne 76.40 ± 0.94 %. Pour les fonctions fenêtrées, la Fig 6.5 résume les taux de classification $\bar{\eta}_{consist}$ de ce deuxième test. Nous constatons que les taux se dégradent d'autant plus que les signaux sont décalés. La disposition des fenêtres introduite dans [MBOL12] n'est donc pas strictement consistante par translation.

6.3 Nouvelle famille de fonctions de goupement

Dans cette section, nous présentons un nouveau fenêtrage, générant une nouvelle famille de fonctions de groupement. Leur consistance par translation sera ensuite testée.

6.3.1 Nouveau fenêtrage

Le problème constaté dans l'expérience précédente vient de la définition de la taille T_H des fenêtres de Hanning qui dépend de N_{max} défini dans l'Eq. (6.5). La valeur N_{max} est sensible à une translation temporelle des signaux car elle ne prend pas en compte le début du support du spikegramme. Dans cette section, nous proposons de calculer la taille de cette fenêtre à partir de l'écart temporel maximum :

$$A_{max} = \max_p \left\{ \max_k \left\{\tau_p^k\right\}_{k=1}^K - \min_k \left\{\tau_p^k\right\}_{k=1}^K + 1 \right\}_{p=1}^P . \tag{6.11}$$

La taille T_H de la fenêtre de Hanning, définie comme $T_H = \frac{A_{max}}{0.5 \times F}$, est donc maintenant indépendante du découpage temporel des signaux. De nouvelles fonctions de groupement, notées $.'$, sont calculées avec ce nouveau fenêtrage. Le déroulé de ce fenêtrage est résumé dans l'Algorithme 15.

Algorithm 15 : $\{\mathsf{v}_p\}_{p=1}^P = .'\left(\left\{\left\{x_{l_p^k,\tau_p^k}\right\}_{k=1}^K\right\}_{p=1}^P\right)$

1: Ecart temporel maximum : $A_{max} \leftarrow \max_p \left\{ \max_k \left\{\tau_p^k\right\}_{k=1}^K - \min_k \left\{\tau_p^k\right\}_{k=1}^K + 1 \right\}_{p=1}^P$
2: Taille de fenêtre : $T_H \leftarrow \frac{A_{max}}{0.5 \times F}$
3: **for** $p \leftarrow 1, P$ **do**
4: Temps du 1er atome : $\tilde{\tau}_p \leftarrow \min_k \left\{\tau_p^k\right\}_{k=1}^K$
5: Calcul de la matrice V_p : Eq. (6.6) ou Eq. (6.9)
6: Vectorisation de la matrice V_p : Eq. (6.7)
7: Normalisation du vecteur v_p : $\mathsf{v}_p \leftarrow \mathsf{v}_p / \|\mathsf{v}_p\|_2$
8: **end for**

6.3.2 Test de consistance

Les nouvelles fonctions de groupement sont utilisées pour l'apprentissage de SVM comme décrit en Section 6.2.1, et avec les mêmes permutations. Les taux de classification $\bar{\eta}$ sont résumés dans la Table 6.2. Nous remarquons que ces taux de classification sont meilleurs que ceux de la Table 6.1. Cette légère différence est due au fait que les signaux originaux de la base UCI *Character Trajectories* ne sont pas tous recalés exactement de la même façon, *i.e.* découpés au même échantillon. Ce léger problème de recalage est aussi constaté dans [VEG12]. Ainsi, une bonne disposition des fenêtres permet de s'affranchir de cet inconvénient lié à l'acquisition de données temporelles. Nous effectuons aussi le test de consistance par translation temporelle, et les taux de classification $\bar{\eta}_{consist}$ sont tous identiques à ceux de la Table 6.2, quelle que soit la valeur de N_Z. Les taux de classification sont inchangés quand les signaux sont translatés, ce qui prouve bien la consistance par translation du nouveau fenêtrage ayant engendré ces fonctions de groupement.

TABLE 6.2 – Taux de classification $\bar{\eta}$ sur les vecteurs de caractéristiques extraits des signaux.

Fonctions fenêtrées	ℓ'_0	$\ell'_{0.5}$	ℓ'_1	ℓ'_2	ℓ'_∞	$\sum'(.)$
Taux $\bar{\eta}$ (%)	60.23 ± 0.38	79.98 ± 0.06	84.81 ± 0.30	80.28 ± 0.94	85.05 ± 0.23	90.88 ± 0.06

Nous traçons en Fig. 6.6 les taux $\bar{\eta}_{consist}$ pour les fonctions $\sum(.)$ et $\sum'(.)$ en fonction de la valeur N_Z. La consistance par translation de la fonction de groupement $\sum'(.)$ est bien mise en valeur. Dans cette expérience, les résultats ont été obtenus sur une grille de paramètres SVM restreinte, à cause du grand nombre de signaux et des multiples fonctions de groupement à tester. Le but n'est pas d'atteindre l'état de l'art, mais d'avoir une base commune pour une comparaison équitable des descripteurs.

Ce test est aussi réalisé avec un réseau de neurones convolutifs [LBBH98], *Convolutional Neural Network* (CNN) en anglais, qui est la méthode de référence pour la classification de signaux temporels (cf. Section 6.4.3 pour plus de détails). Les résultats obtenus sont eux aussi ajoutés à la Fig. 6.6. Le taux de classification est de 98.85 ± 0.33% pour $N_Z = 0$, ce qui est l'état de l'art en matière de classification de signaux temporels. Cependant, nous observons que les performances se dégradent très vite quand les signaux sont translatés. Ce phénomène vient du fait que l'apprentissage du CNN est fait seulement sur des signaux non translatés. Dans ce test de consistance, les signaux sont translatés. Si les fonctions de groupement sont effectivement consistantes par translation, il n'y a pas de modification de la distribution statistique des vecteurs de caractéristiques : les performances du classifieur restent donc identiques. Par contre, quand le classifieur est appliqué sur les signaux bruts comme pour le CNN, la distribution des signaux est modifiée par les translations, et les performances du classifieur se dégradent. Par la suite, nous réaliserons un test complémentaire pour lequel nous introduirons des signaux translatés aléatoirement dans l'ensemble d'apprentissage du CNN.

6.3.3 Discussion

Une fonction de groupement non fenêtrée est donc pleinement consistante par translation. Les expériences réalisées nous ont permis de constater qu'une approche fenêtrée améliore les résultats de classification, tout en rajoutant un hyper-paramètre concernant le nombre de fenêtres. Une fonction de groupement fenêtrée est un compromis entre la consistance par translation et la préservation d'informa-

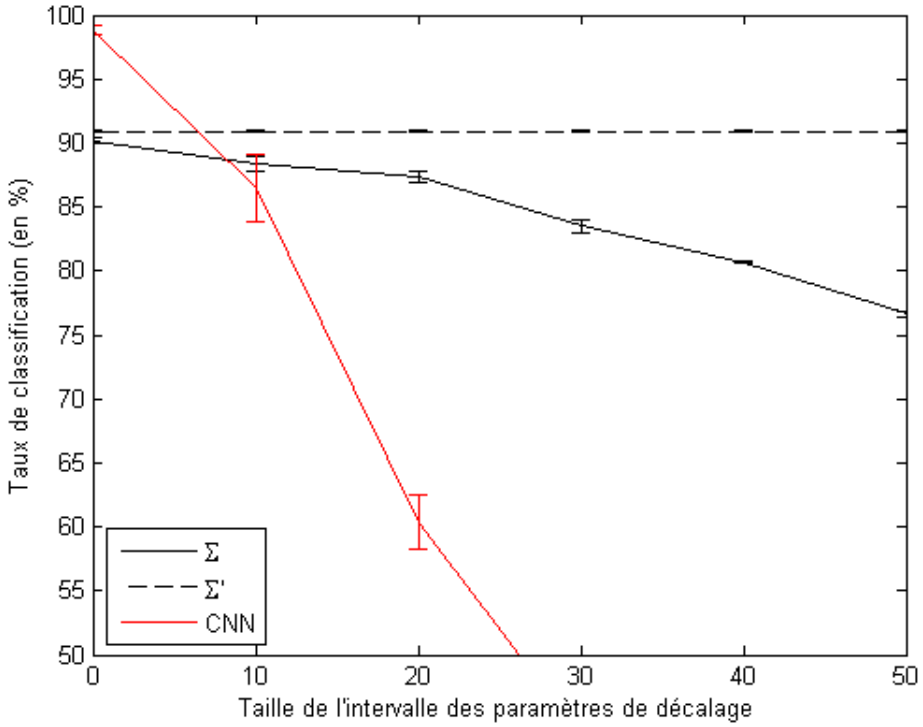


Fig. 6.6 – Taux de classification $\bar{\eta}_{consist}$ sur les vecteurs de caractéristiques extraits des signaux translatés aléatoirement par les fonctions de groupement $\sum(.)$ et $\sum'(.)$, et pour le CNN.

tions relatives à la localisation et à l'ordre des coefficients. S'il n'y a qu'une seule fenêtre, nous retombons sur le cas non fenêtré consistant par translation ; s'il y a autant de fenêtres que d'échantillons, la fonction ne modifie pas le vecteur d'entrée, et toutes les informations d'ordre sont conservées. Une perspective est de tester d'autres types de fenêtres : triangle, Hamming, etc.

Le CNN a de meilleurs résultats que les approches basées sur les descripteurs issus de spikegramme. Cela provient du fait que c'est un classifieur discriminatif, optimisé entièrement grâce à une fonctionnelle de classification. Cependant, la dernière expérience montre que le CNN réclame une segmentation des signaux qui soit toujours identique. Un décalage trop important dans la découpe initiale d'un signal met en défaut le CNN.

Pour conclure, les différentes fonctions de groupement consistantes par translation ont été comparées sur les mêmes données, avec le même classifieur et la même grille de paramètres. Cette comparaison a montré l'optimalité du fenêtrage présenté, générant une nouvelle famille de fonctions de groupement. D'autre part, cette comparaison a montré les avantages (pas de perte de performances lors de translation) et les limites (performances de bases plus faibles) de telles approches par rapport à un classifieur comme le CNN.

6.4 Apprentissage de dictionnaire discriminant

Les décompositions sur un dictionnaire appris permettent des classifications satisfaisantes, mais qui restent limitées par leur caractère génératif. De plus, cette approche est effectuée en deux étapes indépendantes : apprentissage du dictionnaire puis apprentissage du classifieur. Une des façons d'améliorer cette

classification est d'intégrer des contraintes de discrimination lors de l'apprentissage du dictionnaire. Cette section est en réalité une discussion sur l'apprentissage de dictionnaire discriminant, sur l'unification de plusieurs domaines, et sur les perspectives futures.

6.4.1 Contexte

L'apprentissage de dictionnaire est naturellement porté vers le *clustering* non-supervisé (cf. lien avec les K-*means* en Section 1.3.3) et il est parfois utilisé pour faire de la classification [TF11]. Cependant, l'apprentissage du dictionnaire et l'entraînement du classifieur sont effectués indépendamment, ce qui est une perte d'efficacité. De plus, si les décompositions adaptées montrent une certaine reproductibilité (cf. Fig. 6.3), le dictionnaire appris est optimisé seulement par des critères ℓ_2 d'erreur de représentation : il ne possède donc que des propriétés génératives, et non discriminatives. Le but est donc d'apprendre un dictionnaire de manière supervisée, en intégrant des contraintes de discrimination qui renforceront ses performances de classification. Si l'apprentissage de dictionnaire généralise l'algorithme de *clustering* des K-*means*, la décomposition parcimonieuse est instable par nature. Le but d'intégrer des propriétés discriminatives est d'améliorer les performances de classification en stabilisant les décompositions intra-classes.

A ce stade, deux tendances se rejoignent : les approches discriminatives (apprentissage de frontières entre les classes) souhaitent intégrer des propriétés génératives pour être plus robustes au bruit [HA06], et les approches génératives (apprentissage des barycentres des classes) souhaitent intégrer des propriétés discriminatives pour améliorer leurs performances par rapport aux approches discriminatives. Récemment, plusieurs travaux essaient de lier ces deux approches, et les résultats sont prometteurs. Le critère d'optimisation d'un dictionnaire discriminant est un compromis entre reconstruction et discrimination. Si le paramètre de compromis tend vers la reconstruction, nous retrouvons l'apprentissage de dictionnaire classique ; s'il tend vers la discrimination, nous retrouvons un classifieur discriminatif [HTF09].

Nous faisons un état de l'art méthodologique sur les apprentissages de dictionnaires discriminants, puis nous étudierons le cas des dictionnaires invariants par translation.

6.4.2 Cas classique : état de l'art

Nous faisons d'abord la revue des décompositions parcimonieuses discriminantes, puis des apprentissages de dictionnaires discriminants. Ce travail est d'autant plus utile que la plupart de ces travaux sont très récents, et qu'il n'existe pas de synthèse générale sur le sujet. Ces méthodes utilisent parfois les approximations parcimonieuses multicanales pour traiter parallèlement les P signaux groupés dans $Y \in \mathbb{R}^{N \times P}$, et obtenir ainsi les coefficients de décomposition $X \in \mathbb{R}^{M \times P}$ (avec M le nombre d'atomes).

Décompositions parcimonieuses discriminantes

Une des façons les plus répandues est de constituer un dictionnaire Φ concaténant C sous-dictionnaires Φ_c, soit un par classe. De plus, la fonction $\delta_c(.) : \mathbb{R}^M \to \mathbb{R}^M$ sélectionne les coefficients associés à la classe Cl_c et met à zéro tous les autres. La méthode *Sparse Representation based Classification* (SRC) [WYG+09] calcule les coefficients de décomposition $\hat{x}$ sur Φ, puis un résidu est calculé pour chaque classe en utilisant $\Phi\delta_c(\hat{x})$ comme approximation. La classification est effectuée en cherchant la classe qui

minimise ainsi le résidu :

$$\hat{c} = \text{argmin}_c \ \|y - \Phi\delta_c(\hat{x})\|_2^2 \ . \tag{6.12}$$

D'autres méthodes introduisent des termes discriminatifs lors de la sélection des atomes. Un critère de Fisher (variance inter-classe divisée par variances intra-classe) est calculé sur l'ensemble des coefficients de décomposition, puis il est ajouté dans la fonctionnelle d'optimisation d'un OMP multicanal [HA06] : l'algorithme sélectionne donc les atomes selon un compromis entre reconstruction et discrimination. Toujours dans un cas multicanal, [KF08] propose une fonctionnelle de sélection comportant trois termes : une contrainte de ressemblance entre l'atome candidat et la variance inter-classe calculée sur les signaux (et non les coefficients), une contrainte d'orthogonalité entre l'atome candidat et le dictionnaire actif et le terme classique de produit scalaire entre l'atome et le résidu.

Si ces méthodes fournissent des décompositions discriminantes, elles doivent faire le choix d'un dictionnaire *ad hoc*. Les signaux de l'ensemble d'apprentissage peuvent être choisis comme dictionnaire [WYG+09], le meilleur dictionnaire discriminant peut être cherché parmi plusieurs dictionnaires analytiques [SC95] ou encore les paramètres d'ondelettes peuvent être optimisés pour améliorer la classification [SSD03, LGP+08, YR11]. Une autre approche consiste à apprendre le dictionnaire discriminant qui fournit la classification optimale.

Apprentissages de dictionnaires discriminants

Si le but de toutes ces méthodes est d'apprendre un dictionnaire discriminant, les implémentations possibles sont nombreuses. Le but de ce paragraphe est de faire une revue structurée de ces multiples approches.

L'une des idées générales est d'effectuer des groupements d'atomes : un dictionnaire ou un sous-dictionnaire est créé pour chaque classe. L'approche la plus ancienne consiste à apprendre un dictionnaire Φ_c par classe sur les signaux d'apprentissage, puis à décomposer le signal à classer sur chaque dictionnaire appris : celui qui minimise la norme du résidu désigne la classe d'attribution [SH06]. Cette idée est reprise dans une procédure itérative et dans un contexte non-supervisé [SS10] : chaque signal d'apprentissage est décomposé sur chaque dictionnaire Φ_c et est attribué à la classe dont le dictionnaire minimise la norme du résidu, les C dictionnaires sont ensuite mis à jour avec leurs signaux attribués. Gardant l'idée de classer les signaux selon la norme des résidus, [MBP+08] apprend un dictionnaire par classe, avec une étape de décomposition classique et une étape de mise à jour du dictionnaire contenant une fonction *softmax*. Cette fonction assure que le résidu sur le dictionnaire cible est faible, tout en maximisant les résidus obtenus sur les autres. Reprenant l'approche de [SH06], [YZYZ10] apprend C dictionnaires indépendants, puis les fusionne dans un dictionnaire utilisé avec le SRC pour classer les signaux.
Une approche plus récente utilise un dictionnaire global concaténant C sous-dictionnaires. L'une des conséquences est que les sous-dictionnaires peuvent partager des atomes en commun, représentant ainsi des motifs communs à toutes les classes. S'inspirant du travail de [SS10], [RSS10] apprend chaque sous-dictionnaire indépendamment, mais ajoute une contrainte de décorrélation entre les atomes des sous-dictionnaires. En plus d'un critère de Fisher (sur les coefficients), les sous-dictionnaires sont aussi utilisés dans [YZFZ11] pour introduire un critère triple de reconstruction discriminative : il favorise l'utilisation exclusive du sous-dictionnaire de la classe du signal. Le critère résultant est aussi bien utilisé dans l'étape de décomposition que dans l'étape de mise à jour du dictionnaire.

Une autre des idées générales est d'effectuer des groupements de signaux d'une même classe grâce à une parcimonie groupée par norme mixte. Cela favorise l'apprentissage d'atomes qui fournissent des décompositions identiques pour tous les signaux de la classe, évitant ainsi les instabilités des décompositions au sein d'une même classe. Un OMP multicanal est modifié dans [RS07] pour intégrer un critère de Fisher dans l'étape de sélection. Les signaux d'une même classe sont décomposés ensemble par un modèle multicanal, mais le dictionnaire appris est commun à toutes les classes.

Toutes les méthodes évoquées précédemment ont pour but d'apprendre des atomes qui favorisent la classification des signaux décomposés, notamment en apprenant ensuite un classifieur. L'inconvénient de cette approche est de diviser l'apprentissage du dictionnaire discriminant et celui du classifieur en deux problèmes indépendants. Certaines approches apprennent un classifieur en même temps que le dictionnaire. Un critère logistique est utilisé pour apprendre un classifieur linéaire (traitant les coefficients) ou bilinéaire (coefficients et signaux) dans [MBP+09]. L'étape de décomposition est inchangée et l'étape de mise à jour des atomes est réalisée de manière alternée avec l'apprentissage du classifieur. Ce travail est généralisé dans [MBP12] au cas semi-supervisé, aux cas multiclasses, etc. Un critère quadratique est utilisé pour entraîner un classifieur linéaire (traitant les coefficients) lors d'un apprentissage alterné entre le dictionnaire et le classifieur [PV08], puis lors d'un apprentissage simultané [ZL10], puis avec l'introduction des sous-dictionnaires dans le *Label Consistent* K-SVD [JLD11].

Pour conclure cette revue, il existe de multiples méthodes d'apprentissage pour intégrer des contraintes discriminatives. La difficulté de cette synthèse est due aux nombreux paramètres qui peuvent être combinés : sous-dictionnaires ou non, contrainte de décorrélation des atomes, choix du critère de discrimination et savoir s'il est appliqué aux signaux, aux coefficients ou aux résidus, intégration du critère discriminant dans la sélection et/ou dans la mise à jour du dictionnaire, etc.[2] De plus, certaines questions restent ouvertes. Dans le cas de sous-dictionnaires, [RSS10] ignorent les atomes communs partagés. Est-ce un avantage ou un inconvénient que les sous-dictionnaires partagent des atomes communs ?

6.4.3 Cas invariant par translation

Notre problème est de savoir comment étendre ce type de méthodes au cas d'invariance par translation. Pour cela, nous évoquons deux méthodes proches de cette problématique.

Lien entre DLA invariant par translation et CNN

Dans ce paragraphe, nous nous intéressons aux réseaux de neurones convolutifs [LBBH98], soit *Convolutional Neural Network* (CNN) en anglais, dédiés à la classification de signaux robuste à la translation. Les CNN sont éminemment efficaces pour la reconnaissance d'écriture mais sont souvent considérés comme des boîtes noires. Leur fonctionnement interne paraît obscur. Nous faisons ici un parallèle entre l'apprentissage de dictionnaire invariant par translation et les CNN, afin de comprendre les ressemblances et les différences.

Dans un premier temps, nous rappelons le fonctionnement du réseau de neurones classiquement utilisé pour la classification, à savoir le Perceptron Multicouches, soit en anglais *Multilayer Perceptron* (MLP) [DHS01]. Inspiré de l'architecture du cerveau, il est composé de plusieurs couches de neurones : la couche d'entrée contenant le vecteur de caractéristiques v_p à classer, une ou plusieurs couches cachées reliées

2. Chacun de ces paramètres serait une façon de structurer une synthèse sur le sujet.

par les pondérations W, et une couche de sortie $z \in \mathbb{R}^C$ composée de $C = 20$ neurones, dont le résultat permet de classer le vecteur d'entrée v_p.
Utilisé comme un classifieur, le MLP est capable d'apprendre de manière supervisée des frontières non-linéaires entre les différentes classes de l'ensemble d'apprentissage $\mathbf{V}_a$, grâce à la mise à jour des pondérations W par rétro-propagation [DHS01] qui est une optimisation non-convexe. Les pondérations sont apprises sur l'ensemble d'apprentissage $\{\mathsf{v}_p\}_{p=1}^P$ de telle sorte que la valeur z_c du $c^{\text{ième}}$ neurone de la couche de sortie z soit égale à 1 si la classe du vecteur d'entrée v_p est la classe Cl_c, pendant que les autres neurones de sortie sont égaux à -1. Ainsi, le MLP minimise l'erreur quadratique E entre les sorties $z \in \mathbb{R}^C$ du MLP et les valeurs cibles $\mathsf{t} \in \mathbb{R}^C$ (aussi appelées labels) selon :

$$E(W) = \sum_{p=1}^{P} (z(\mathsf{v}_p) - \mathsf{t}(\mathsf{v}_p))^2 = \|Z - \mathsf{T}\|_F^2 \ , \tag{6.13}$$

$$\hat{W} = \arg\min_W E(W) \ . \tag{6.14}$$

avec $Z \in \mathbb{R}^{C \times P}$ les sorties du MLP et $\mathsf{T} \in \mathbb{R}^{C \times P}$ les valeurs cibles pour l'ensemble d'apprentissage.

Le CNN est une extension du MLP au cas convolutif [LBBH98]. C'est un système qui traite directement les signaux bruts de l'ensemble d'apprentissage $\boldsymbol{Y}_a$ grâce à une cascade de filtres convolutifs et de sous-échantillonnages effectués par fonctions de groupement fenêtrées. La sortie de cette cascade est ensuite reliée à un MLP classique qui optimise la même fonction de coût que précédemment. Les filtres convolutifs $\tilde{W}$ sont eux aussi appris par rétro-propagation, et optimisés pour extraire automatiquement des signaux $\boldsymbol{Y}_a$ les caractéristiques $\mathbf{V}_a$ favorisant la classification du MLP. Ce système est donc robuste à d'éventuelles translations des signaux d'apprentissage, *à condition* qu'elles restent dans l'intervalle des fenêtres des fonctions de groupement du CNN.

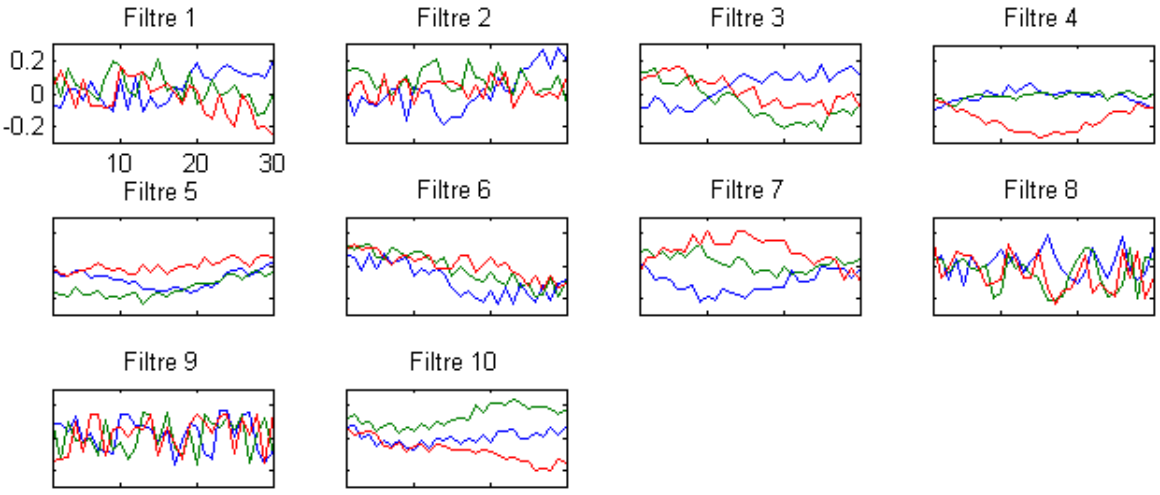


Fig. 6.7 – Filtres convolutifs appris par le CNN sur les signaux bruts.

Nous mettons en place un CNN constitué d'une couche de $L = 10$ filtres convolutifs suivie d'une fonction de groupement fenêtrée effectuant un sous-échantillonnage (moyenne sur 3 éléments), puis d'un MLP à 3 couches. Nous appliquons ce CNN sur les données *Character Trajectories*, et nous affichons les filtres convolutifs appris $\tilde{W}$ en Fig. 6.7. Remarquons que ces filtres sont eux aussi trivariés. Ces filtres

sont difficilement interprétables car ils ne représentent rien par rapport aux signaux : ils sont appris afin d'extraire les caractéristiques les plus discriminantes. Notons aussi que le taux de classification obtenu par le CNN et moyenné sur 10 tests est de $98.85 \pm 0.33\%$.
Pour résumer, le CNN apprend des filtres translatables (Fig. 6.7) qui discriminent/classent au mieux les signaux alors que le DLA invariant par translation apprend des noyaux translatables (Fig. 6.4) qui reconstruisent/représentent au mieux les signaux.

Nous complétons l'expérience de consistance par translation de la Section 6.3.2, dans laquelle nous avons constaté que les taux de classification du CNN se dégradent quand les signaux de test sont translatés. Dans cette expérience, nous translatons aléatoirement les signaux de l'ensemble de test *et* de l'ensemble d'apprentissage. Les taux de classification moyennés sur 10 tests sont affichés en Fig. 6.8. Les valeurs de l'avant de la surface (pas de décalage des signaux d'apprentissage) correspondent aux taux de classification du CNN de la Fig. 6.6. Nous observons que le CNN devient robuste aux translations quand les signaux d'apprentissage sont eux aussi translatés, et qu'il a d'excellents taux de classification. Cette expérience souligne l'avantage des descripteurs consistants par translation qui s'affranchissent du besoin d'introduire les variabilités dans l'ensemble d'apprentissage. Néanmoins, il est plus facile d'introduire des variabilités à valeurs discrètes (décalages τ) qu'à valeurs continues (par exemple, la rotation des trajectoires).

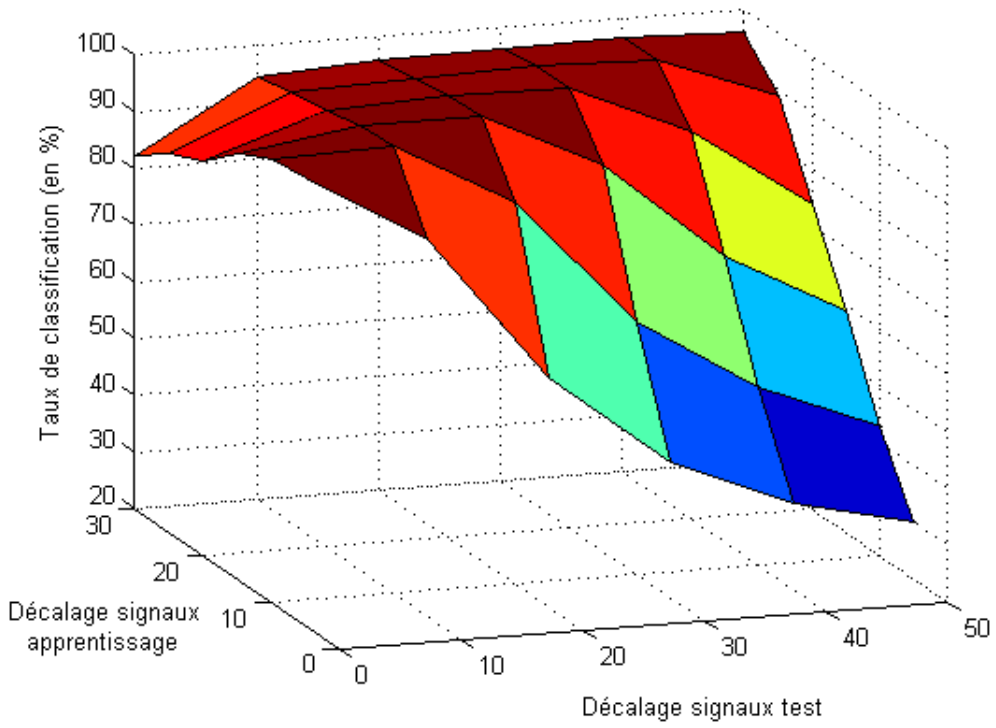


Fig. 6.8 – Taux de classification obtenus pour le CNN en fonction des décalages temporels aléatoires des signaux de test et des signaux d'apprentissage.

Lien entre DLA invariant par translation et la *Scattering Transform*

La *Scattering Transform* introduite dans [Mal12] fournit des représentations ayant des invariances telles que translation, rotation, échelle, etc. Cette transformée est une cascade de convolutions dont l'architecture est proche de celle des CNN. Les filtres de convolutions sont définis comme des ondelettes complexes, et non appris comme dans l'apprentissage de dictionnaire ou dans le CNN. Les convolutions sont suivies d'un opérateur module et d'un filtrage passe-bas réalisé par convolution avec une Gaussienne (ce qui revient à appliquer une norme ℓ_1 pondérée sur une fenêtre glissante), et le vecteur résultant est sous-échantillonné. Au final, cela revient à appliquer une fonction de groupement sur les convolutions. La *Scattering Transform* effectue une décomposition des signaux dans un but reconstructif, similaire à la décomposition sur un dictionnaire invariant par translation, et non dans un but discriminatif comme le CNN.

Les propriétés d'invariances de la décomposition sont éminemment efficaces pour faire de la classification puisqu'elles éliminent des représentations la plupart des variabilités entre les occurrences d'une même classe [BM13]. Les coefficients de décomposition sont utilisés dans des classifieurs et obtiennent des résultats en reconnaissance d'écriture équivalents à ceux du CNN.

Pour conclure cette section, il existe plusieurs méthodes d'apprentissage de dictionnaires discriminants récemment mises au point. Les choix de paramètres étant multiples, elles ont toutes des implémentations différentes. Le domaine est très actif, mais le cas d'invariance par translation n'a pas encore été étudié, ce qui donne une perspective forte aux travaux. Enfin, nous avons fait le lien avec d'autres méthodes voisines, comme le CNN ou la *Scattering Transform*.

Conclusion

Dans ce chapitre, nous avons fait la revue puis comparé les différentes fonctions de groupement avec le même classifieur et les mêmes données. Consistantes par translation, ces fonctions de groupement sont appliquées aux décompositions invariantes par translation. Elles sont inchangées quand les signaux sont translatés, ce qui permet de maintenir une classification identique. Cependant, les résultats ne sont pas aussi bons que les classifieurs du type CNN.

Pour améliorer la classification de ces décompositions parcimonieuses, il est possible d'apprendre un dictionnaire discriminant. Ce domaine est en plein essor, et de multiples méthodes existent. Les perspectives sont de mettre en œuvre un apprentissage de dictionnaire discriminant et invariant par translation.

Bibliographie

[BM13] J. Bruna et S. Mallat : Invariant scattering convolution networks. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35:1872–1886, 2013.

[BMB+09] J.P. Bandera, R. Marfil, A. Bandera, J.A. Rodriguez, L. Molina-Tanco et F. Sandoval : Fast gesture recognition based on a two-level representation. *Pattern Recognition Letters*, 30:1181–1189, 2009.

[CAS05] A. Croitoru, P. Agouris et A. Stefanidis : 3D trajectory matching by pose normalization. *In Proc. ACM Int. Workshop on Geographic Information Systems GIS '05*, pages 153–162, 2005.

[DHS01] R.O. Duda, P.E. Hart et D.G. Stork : *Pattern Classification*. 2nd edition, Wiley-Interscience, 2001.

[GRKN07] H. Grosse, R. Raina, R. Kwong et A.Y. Ng : Shift-invariant sparse coding for audio classification. *In Proc. Conf. Uncertainty in Artificial Intelligence UAI*, pages 149–158, 2007.

[HA06] K. Huang et S. Aviyente : Sparse representation for signal classification. *In Advances in Neural Information Processing Systems NIPS*, pages 609–616, 2006.

[HTF09] T. Hastie, R. Tibshirani et J. Friedman : *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. Springer New York Inc., 2009.

[HYHJ+12] P.-S. Huang, J. Yang, M. Hasegawa-Johnson, F. Liang et T.S. Huang : Pooling robust shift-invariant sparse representations of acoustic signals. *In Conf. Int. Speech Communication Association, Interspeech 2012*, 2012.

[JLD11] Z. Jiang, Z. Lin et L.S. Davis : Learning a discriminative dictionary for sparse coding via label consistent K-SVD. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2011*, pages 1697–1704, 2011.

[KF08] E. Kokiopoulou et P. Frossard : Semantic coding by supervised dimensionality reduction. *IEEE Trans. on Multimedia*, 10:806–818, 2008.

[LBBH98] Y. LeCun, L. Bottou, Y. Bengio et P. Haffner : Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

[LeC12] Y. LeCun : Learning invariant feature hierarchies. *In Computer Vision – ECCV 2012*, volume 7583 de *Lecture Notes in Computer Science*, pages 496–505, 2012.

[LGP+08] M.-F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli et D. Farina : Multi-channel surface EMG classification using support vector machines and signal-based wavelet optimization. *Biomedical Signal Processing and Control*, 3:169–174, 2008.

[Mal12] S. Mallat : Group invariant scattering. *Commun. Pure Appl. Math.*, 65:1331–1398, 2012.

[MBM+12] A. Mouraud, Q. Barthélemy, A. Mayoue, C. Gouy-Pailler, A. Larue et H. Paugam-Moisy : From neuronal cost-based metrics to sparse coded signals classification. *In Proc. Eur. Symp. Artificial Neural Networks, Computational Intelligence and Machine Learning ESANN*, pages 311–316, 2012.

[MBOL12] A. Mayoue, Q. Barthélemy, S. Onis et A. Larue : Preprocessing for classification of sparse data : Application to trajectory recognition. *In Proc. IEEE Workshop on Statistical Signal Processing SSP '12*, pages 37–40, 2012.

[MBP+08] J. Mairal, F. Bach, J. Ponce, G. Sapiro et A. Zisserman : Discriminative learned dictionaries for local image analysis. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008*, pages 1–8, 2008.

[MBP+09] J. Mairal, F. Bach, J. Ponce, G. Sapiro et A. Zisserman : Supervised dictionary learning. *In Advances in Neural Information Processing Systems*, pages 1033–1040, 2009.

[MBP12] J. Mairal, F. Bach et J. Ponce : Task-driven dictionary learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34:791–804, 2012.

[PV08] D.S. Pham et S. Venkatesh : Joint learning and dictionary construction for pattern recognition. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008*, pages 1–8, 2008.

[RS07] F. Rodriguez et G. Sapiro : Sparse representations for image classification : Learning discriminative and reconstructive non-parametric dictionaries. Rapport technique IMA Preprint 2213, University of Minnesota, December 2007.

[RSS10] I. Ramirez, P. Sprechmann et G. Sapiro : Classification and clustering via dictionary learning with structured incoherence and shared features. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2010*, pages 3501–3508, 2010.

[SC95] N. Saito et R.R. Coifman : Local discriminant bases and their applications. *Journal of Mathematical Imaging and Vision*, 5:337–358, 1995.

[SH06] K. Skretting et J.H. Husøy : Texture classification using sparse frame based representation. *EURASIP J. on Applied Signal Processsing*, 2006:1–11, 2006.

[SP11] S. Scholler et H. Purwins : Sparse approximations for drum sound classification. *IEEE Journal of Selected Topics in Signal Processing*, 5:933–940, 2011.

[SS10] P. Sprechmann et G. Sapiro : Dictionary learning and sparse coding for unsupervised clustering. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP 2010*, pages 2042–2045, 2010.

[SSD03] D.J. Strauss, G. Steidl et W. Delb : Feature extraction by shape-adapted local discriminant bases. *Signal Process.*, 83:359–376, 2003.

[TF11] I. Tošić et P. Frossard : Dictionary learning. *IEEE Signal Processing Magazine*, 28:27–38, 2011.

[VEG12] C. Vollmer, J.P. Eggert et H.-M. Gross : Modeling human motion trajectories by sparse activation of motion primitives learned from unpartitioned data. *In KI 2012 : Advances in Artificial Intelligence*, volume 7526 de *Lecture Notes in Computer Science*, pages 168–179, 2012.

[WYG+09] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry et Y. Ma : Robust face recognition via sparse representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31:210–227, 2009.

[YR11] F. Yger et A. Rakotomamonjy : Wavelet kernel learning. *Pattern Recognition*, 44:2614–2629, 2011.

[YZFZ11] M. Yang, L. Zhang, X. Feng et D. Zhang : Fisher discrimination dictionary learning for sparse representation. *In Proc. Int. Conf. Computer Vision ICCV '11*, pages 543–550, 2011.

[YZYZ10] M. Yang, L. Zhang, J. Yang et D. Zhang : Metaface learning for sparse representation based face recognition. *In Proc. IEEE Int. Conf. Image Processing ICIP 2010*, pages 1601–1604, 2010.

[ZL10] Q. Zhang et B. Li : Discriminative K-SVD for dictionary learning in face recognition. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2010*, pages 2691–2698, 2010.

Conclusion et perspectives

Conclusion

Dans cette thèse, nous nous sommes intéressés aux méthodes d'approximation et d'apprentissage qui fournissent des représentations parcimonieuses et nous avons effectué une revue de ce domaine. Ces méthodes permettent d'analyser des bases de données très redondantes à l'aide de dictionnaires d'atomes appris. Etant adaptés aux données étudiées, ils sont plus performants en qualité de représentation que les dictionnaires classiques dont les atomes sont définis analytiquement.

Nous avons considéré plus particulièrement des signaux multivariés résultant de l'acquisition simultanée de plusieurs grandeurs, comme les signaux EEG ou les signaux de mouvements 2D et 3D. Nous avons étendu les méthodes de représentations parcimonieuses au modèle multivarié, pour prendre en compte les interactions entre les différentes composantes acquises simultanément. Ce modèle est plus flexible que l'habituel modèle multicanal qui impose une hypothèse de rang 1. Pour les EEG, nous avons constaté que le dictionnaire appris multivarié est plus efficace que le dictionnaire de Gabor multicanal et, en apprenant un motif du potentiel évoqué P300, nous avons montré que le dictionnaire appris peut avoir une signification physiologique.

Nous avons étudié des modèles de représentations invariantes : invariance par translation temporelle, invariance par rotation, etc. En ajoutant des degrés de liberté supplémentaires, chaque noyau est potentiellement démultiplié en une famille d'atomes, translatés à tous les échantillons, tournés dans toutes les orientations, etc. Ainsi, un dictionnaire de noyaux invariants génère un dictionnaire d'atomes très redondant, et donc idéal pour représenter les données étudiées redondantes. Toutes ces invariances nécessitent la mise en place de méthodes adaptées à ces modèles.

L'invariance par translation temporelle est une propriété incontournable pour l'étude de signaux temporels ayant une variabilité temporelle naturelle. Nous avons illustré cela sur les signaux EEG : le potentiel évoqué P300 possède une latence temporelle que notre méthode prend en compte contrairement aux approches classiques de moyennage.

Dans le cas de l'invariance par rotation 2D, le passage en complexe suffit à résoudre le problème. Dans le cas 3D, l'extension aux quaternions ne permet pas de résoudre le problème d'invariance par rotation. Nous utilisons une méthode basée sur le recalage de Procrustes. Nous avons constaté l'efficacité de l'approche non-orientée sur celle orientée, même dans le cas où les données ne sont pas tournées. En effet, le modèle non-orienté permet de détecter les invariants des données, et assure la robustesse à la rotation quand les données tournent. Nous avons illustré ce phénomène sur des données de mouvements 2D et 3D.

Nous avons aussi constaté la reproductibilité des décompositions parcimonieuses sur un dictionnaire appris. Cette propriété générative s'explique par le fait que l'apprentissage de dictionnaire est une généralisation des K-*means*. D'autre part, nos représentations possèdent de nombreuses invariances, ce qui est idéal pour faire de la classification.

Nous avons donc étudié comment effectuer une classification adaptée au modèle d'invariance par translation, en utilisant des fonctions de groupement consistantes par translation. Nous avons fait une comparaison des différentes fonctions de l'état de l'art avec le même classifieur et les mêmes données, et nous avons proposé un nouveau fenêtrage. Les performances restent moins bonnes que les classifieurs discriminatifs tels que les CNN, ce qui ouvre des perspectives d'amélioration.

Perpectives

Dans cette dernière partie, nous évoquerons plusieurs perspectives concernant les méthodes de décomposition, les modèles d'invariance ou encore les liens entre décomposition et classification.

Les algorithmes de décomposition présentés dans cette thèse sont basés sur l'OMP qui est une méthode parcimonieuse ℓ_0 non-convexe. Cette non-convexité peut bloquer l'optimisation dans un minimum local, et favoriser ainsi l'instabilité naturelle des décompositions parcimonieuses. Ce phénomène a été illustré en Section 4.5.2 avec l'exemple de la lettre v. La première des perspectives est de « convexifier » les algorithmes présentés pour améliorer leurs résultats, tout en prenant en compte le fait qu'ils doivent être implémentés dans un cadre d'invariance par translation temporelle.

Nous avons aussi introduit plusieurs invariances dans les modèles de décomposition : à l'échelle (ou dilatation spatiale), à la translation temporelle, à la rotation, etc. Une des perspectives est d'intégrer dans notre modèle l'invariance par dilatation temporelle. Dans le cas de l'étude de gestes, cela permettrait d'analyser le même mouvement, réalisé lentement ou rapidement, avec les mêmes noyaux. En ajoutant un degré de liberté supplémentaire, le noyau :

$$\psi_l(t-\tau) \quad \text{devient} \quad \frac{1}{\sqrt{s}} \cdot \psi_l\left(\frac{t-\tau}{s}\right) ,$$

avec s le paramètre de dilatation temporelle, choisi sur une grille uniforme ou dyadique. Un modèle utilisant de tels noyaux est robuste à la dilatation temporelle des signaux.

Nous rappelons le modèle multivarié généralisé (2.17) présenté au Chapitre 2. En considérant le signal multivarié $\boldsymbol{y} \in \mathbb{R}^{N \times V}$, les atomes multivariés $\boldsymbol{\phi}_m \in \mathbb{R}^{N \times U_m}$ ayant chacun leur dimension spécifique U_m, et les matrices $R_m \in \mathbb{R}^{U_m \times V}$, le modèle s'écrit :

$$\boldsymbol{y} = \sum_{m=1}^{M} x_m \boldsymbol{\phi}_m R_m + \boldsymbol{\epsilon} \, .$$

Dans cette thèse, nous nous sommes restreints au cas où $\forall m \in \mathbb{N}_M, U_m = V$, et nous avons traité les cas où les matrices $R_m \in \mathbb{R}^{V \times V}$ sont des matrices de rotation pour V quelconque. Les perspectives sont de relâcher ces hypothèses fortes. Dans un premier temps, nous pouvons considérer des matrices $R_m \in \mathbb{R}^{U_m \times V}$ de rang plein : dans ce cas, le modèle devient invariant par combinaison linéaire. Dans un deuxième temps, nous pouvons relâcher toutes les hypothèses, et traiter le modèle général (6.15).

Dans les Chapitres 4 et 5, nous avons évoqué le cas de P capteurs faisant l'acquisition simultanée de signaux (synchronisation temporelle) n'étant pas soumis à la même rotation. Notons que ces P capteurs considérés n'ont pas forcément le même nombre de composantes : ils peuvent acquérir des signaux univariés, bivariés et trivariés (par exemple, mouvement 2D bivarié et pression univariée étudiés en Chapitre 6). Le signal multicapteurs est donc composé de P blocs.

Deux cas sont possibles : soit les atomes des P blocs sont indépendants, alors dans ce cas, les P signaux sont traités isolément. Soit les atomes des P blocs ne sont pas indépendants, *i.e.* un motif caractéristique présent dans un des blocs se répète systèmatiquement et de manière simultanée avec des motifs caractéristiques des autres blocs. Alors, il est possible d'apprendre des atomes multicapteurs composés de P blocs, reliés par la simultanéité temporelle, mais chacun étant libre de son orientation.

L'algorithme de décomposition d'un tel modèle recale d'abord chaque bloc selon son orientation optimale (bloc univarié : signe du coefficient ; bloc bivarié : argument du coefficient complexe ; bloc trivarié : matrice de rotation), puis effectue la somme des P coefficients de corrélation recalés, et choisit l'échantillons τ qui maximise la somme. L'algorithme itère après avoir supprimé du résidu la contribution de l'atome sélectionné. Une des perspectives est donc d'étudier et d'appliquer cet algorithme que nous pouvons qualifier d'invariant à la rotation par bloc (*Block Rotation Invariant* (BRI)) et qui est utile pour traiter des signaux issus de plusieurs capteurs n'étant pas soumis à la même rotation.

Le Chapitre 6 décrit une méthode qui permet de faire de la classification de signaux temporels en prenant en compte l'invariance par translation temporelle. La première perspective, bien détaillée en fin de chapitre, est d'intégrer des contraintes de discrimination dans l'apprentissage de dictionnaires invariants par translation, afin que les décompositions soient plus facilement classifiables.

La deuxième perspective concernant ce chapitre est d'étudier comment traiter l'invariance par rotation en intégrant la valeur des angles dans les vecteurs de caractéristiques. Actuellement, l'orientation des noyaux n'est pas prise en compte dans les descripteurs, ce qui peut dégrader les performances dans les cas ambigus où deux signaux différents utilisent les mêmes noyaux mais avec des orientations différentes. Aussi, il est nécessaire de travailler avec des angles relatifs à la ligne de base (orientation principale des mots, dans le cas de l'écriture manuscrite) et d'intégrer la valeur des angles dans le processus de classification. De plus, il faut que ce procédé puisse autant traiter les orientations 2D que 3D.

Chapitre 7

Annexes

7.1 Complément pour la Section 1.4.2

Nous cherchons à démontrer que : $f(\theta) = \theta^2/2 + \cos(\theta) - 1 \geq 0$. D'une part, la dérivée vaut : $f'(\theta) = \theta - \sin(\theta)$, et la dérivée seconde : $f''(\theta) = 1 - \cos(\theta) \geq 0$. Comme la dérivée seconde f'' est à valeurs positives ou nulles, nous en déduisons que f est une fonction convexe. D'autre part, le minimum de f est atteint pour $f'(\theta) = 0$, soit en $\theta_{min} = 0$, et nous avons $f(\theta_{min}) = 0$. Au final, f est une fonction convexe, et donc toujours supérieure à son minimum égal à 0. Nous en déduisons donc que $f(\theta) \geq 0$. Par conséquent : $1 - \cos(\theta) \leq \theta^2/2$.

7.2 Considérations sur l'implémentation du M-OMP

Nous allons détailler la complexité du M-OMP dans le cas d'invariance par translation. Souvent, les signaux acquis sont dyadiques (*i.e.* la longueur N des signaux est une puissance de 2). Si ce n'est pas le cas, ils sont complétés avec des zéro pour avoir $\lceil N \rceil$ échantillons, avec $\lceil N \rceil$ la première puissance de 2 supérieure à N. Ainsi, dans le cas univarié, la corrélation est calculée par FFT en $O(\lceil N \rceil \log \lceil N \rceil)$ pour chaque noyau. Dans le cas multivarié, la corrélation multivariée est la somme des V corrélations de composantes, et elle est calculée en $O(V \cdot \lceil N \rceil \log \lceil N \rceil)$ pour chaque noyau.

Pour retrouver le cas classique, nous pouvons simplement vectoriser le signal de $N \times V$ en $NV \times 1$. Cependant, une complétion de zéro est nécessaire entre les différentes composantes, sinon les composantes du noyau peuvent recouvrir les composantes consécutives du signal lors du calcul de la corrélation. En limitant la longueur maximale des noyaux à N_L échantillons (ce qui est une perte de flexibilité) avec N_L la longueur du plus long noyau, une complétion de zéro de N_L échantillons doit être effectuée entre deux composantes successives.

Le signal complété de $V(N + N_L)$ échantillons est encore allongé afin d'être dyadique. Finalement, la compléxité de la corrélation est $O\Big((\lceil V(N+N_L) \rceil) \log(\lceil V(N+N_L) \rceil)\Big)$ pour chaque noyau. De plus, pour l'étape de sélection, la recherche du maximum doit être limitée aux $N+N_L$ premiers échantillons de la corrélation obtenue.

Pour conclure, le modèle multivarié est plus facile à implémenter pour le M-OMP et a une complexité plus faible que le modèle classique avec les variables vectorisées.

7.3 Calcul du Hessien

Dans cette annexe, nous expliquons le calcul du Hessien H_l. Cela permet de spécifier le pas adaptatif λ à chaque noyau ψ_l, et de stabiliser la convergence des noyaux très utilisés lors des premières itérations de l'apprentissage. Le calcul du Hessien est donné ici dans le cas de signaux complexes.

Un Hessien moyen H_l est calculé pour chaque noyau ψ_l, et non en chaque échantillon, pour éviter les fluctuations entre les échantillons voisins : H_l est donc réduit à un scalaire. En faisant l'hypothèse de parcimonie (peu d'atomes sont utilisés pour l'approximation), les recouvrements des atomes sélectionnés sont initialement considérés comme inexistants. Ainsi, les termes des dérivées croisées de H_l sont nuls, et nous avons :

$$H_l^i = \sum_{\tau \in \sigma_l} \left| x_{l,\tau}^i \right|^2 . \tag{7.1}$$

Pour les atomes qui se recouvrent, la méthode d'apprentissage peut devenir instable à cause de l'erreur faite dans l'estimation du gradient. Nous surestimons légèrement le Hessien H_l pour compenser cela. Tous les $\tau \in \sigma_l$ sont classés par ordre croissant et indéxés par j tels que : $\tau_1 < \tau_2 ... < \tau_j < \tau_{j+1} ... < \tau_{|\sigma_l|}$, avec $|\sigma_l|$ le cardinal de l'ensemble σ_l. En notant T_l^i la longueur du noyau ψ_l à l'itération i, l'ensemble J_l est défini comme : $J_l = \left\{ j \in \mathbb{N}_{|\sigma_l - 1|} : \tau_{j+1} - \tau_j < T_l^i \right\}$. Cela permet d'identifier les situations de recouvrements. Les termes des dérivées croisées de H_l ne sont plus considérés nuls, et leurs contributions sont proportionnelles à $x_{l,\tau_j}^{i\,*}\, x_{l,\tau_{j+1}}^i + x_{l,\tau_j}^i\, x_{l,\tau_{j+1}}^{i\,*} = 2\,\Re(x_{l,\tau_j}^{i\,*}\, x_{l,\tau_{j+1}}^i)$. Les situations de doubles recouvrements (ou plus) ne sont pas prises en compte, *i.e.* quand $\tau_{j+2} - \tau_j < T_l^i$. Grâce à l'hypothèse de parcimonie, ces situations sont considérées comme très rares (ce qui est vérifié en pratique), et sont compensées en surestimant H_l. Nous prenons la valeur absolue des termes croisés : $\left| x_{l,\tau_j}^{i\,*}\, x_{l,\tau_{j+1}}^i \right| \geq 2\Re(x_{l,\tau_j}^{i\,*}\, x_{l,\tau_{j+1}}^i)$. Cette valeur absolue n'est pas dérangeante, même avec des recouvrements multiples, parce qu'il est mieux de surestimer légèrement H_l que de le sous-estimer (« il vaut mieux avancer peu mais sûrement »). Finalement, nous proposons l'approximation suivante pour H_l, rapide à calculer :

$$H_l^i = \sum_{\tau \in \sigma_l} \left| x_{l,\tau}^i \right|^2 + 2 \sum_{j \in J_l} \frac{T_l^i - (\tau_{j+1} - \tau_j)}{T_l^i} \left| x_{l,\tau_j}^{i\,*}\, x_{l,\tau_{j+1}}^i \right| . \tag{7.2}$$

Quelques commentaires peuvent être faits en regardant l'Eq. (7.2) :

- si l'écart entre les supports de deux atomes est toujours plus grand que T_l, nous retrouvons la première approximation de l'Eq. (7.1) ;
- quand le recouvrement est faible, les produits croisés influencent peu le Hessien ;
- les recouvrements intra-noyau ont été considérés, mais pas ceux inter-noyaux. Cependant, nous observons que ces recouvrements inter-noyaux ne perturbent pas l'apprentissage, donc nous faisons le choix de les ignorer.

L'étape de mise à jour basée sur l'Eq. (2.21) et l'Eq. (7.2) est appelée Mise à jour LM (étape 5, Algorithme 7).

Sans le Hessien dans l'Eq. (2.21), nous retrouvons une mise à jour du 1$^{\text{er}}$ ordre. Dans ce cas, la vitesse de convergence est directement liée à la somme des coefficients de décomposition. L'avantage du Hessien est de tendre à rendre la vitesse de convergence similaire pour tous les noyaux, indépendamment de leurs utilisations dans les décompositions. Concernant l'approximation du Hessien, au début de l'apprentissage, les noyaux qui sont encore des bruits blancs se recouvrent fréquemment et la méthode peut devenir

instable. En augmentant le Hessien, l'approximation stabilise donc le début du processus d'apprentissage. Après, puisque les noyaux convergent, les recouvrements deviennent plus rares et l'approximation du Hessien devient proche de l'Eq. (7.1).

7.4 Produits scalaires pour les quaternions

L'algèbre des quaternions étant non-commutative, nous devons choisir un modèle de décomposition linéaire avec multiplication des scalaires soit à droite, soit à gauche. Nous souhaitons définir un produit scalaire associé à chacun des modèles. La non-commutativité nous impose de relâcher la définition du produit scalaire hermitien, *i.e.* défini positif et sesquilinéaire (*i.e.* linéarité par rapport à un des arguments, semi-linéarité par rapport à l'autre).

Dans ce paragraphe, nous allons étudier le modèle quaternionique linéaire avec multiplication des scalaires à droite : $y = \sum_{m=1}^{M} \phi_m \, x_m + \epsilon$.

Définition

Une application de $\mathbb{H}^N \times \mathbb{H}^N \to \mathbb{H}$ *est un* produit scalaire hermitien à droite *si elle est définie positive et sesquilinéaire à droite (i.e. linéarité par rapport au 1^er argument, semi-linéarité par rapport au 2^ième argument).*

Proposition

Soient les vecteurs quaternioniques $\mathfrak{q}_1$ *et* $\mathfrak{q}_2 \in \mathbb{H}^N$*, l'application :*

$$\begin{aligned} &\mathbb{H}^N \times \mathbb{H}^N \to \mathbb{H} \\ &(\mathfrak{q}_1, \mathfrak{q}_2) \mapsto \langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_{\mathfrak{r}} = \mathfrak{q}_2^H \mathfrak{q}_1 \, , \end{aligned}$$

est un produit scalaire hermitien à droite.

Preuve

Nous considérons les vecteurs $\mathfrak{q}_1$, $\mathfrak{q}_2$ et $\mathfrak{q}_3 \in \mathbb{H}^N$ et les scalaires λ et $\mu \in \mathbb{H}$. L'application proposée ci-dessus est :

1 - définie : $\langle \mathfrak{q}_1, \mathfrak{q}_1 \rangle_{\mathfrak{r}} = 0 \;\Rightarrow\; \mathfrak{q}_1 = 0$;

2 - positive : $\langle \mathfrak{q}_1, \mathfrak{q}_1 \rangle_{\mathfrak{r}} = \mathfrak{q}_1^H \mathfrak{q}_1 = \sum_{n=1}^{N} \mathfrak{q}_1(n)^2 \in \mathbb{H}^+$;

3 - symétrique hermitienne : $\langle \mathfrak{q}_2, \mathfrak{q}_1 \rangle_{\mathfrak{r}} = \mathfrak{q}_1^H \mathfrak{q}_2 = (\mathfrak{q}_2^H \mathfrak{q}_1)^* = \langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_{\mathfrak{r}}^*$;

4 - linéaire à droite par rapport au 1^er argument :
$\langle \mathfrak{q}_1 \lambda + \mathfrak{q}_2 \mu, \mathfrak{q}_3 \rangle_{\mathfrak{r}} = \mathfrak{q}_3^H \mathfrak{q}_1 \lambda + \mathfrak{q}_3^H \mathfrak{q}_2 \mu = \langle \mathfrak{q}_1, \mathfrak{q}_3 \rangle_{\mathfrak{r}} \, \lambda + \langle \mathfrak{q}_2, \mathfrak{q}_3 \rangle_{\mathfrak{r}} \, \mu$;

5 - semi-linéaire à droite par rapport au 2ième argument :
$\langle \mathfrak{q}_1, \mathfrak{q}_2\lambda + \mathfrak{q}_3\mu \rangle_\mathsf{r} = (\mathfrak{q}_2\lambda)\,{}^H\mathfrak{q}_1 + (\mathfrak{q}_3\mu)\,{}^H\mathfrak{q}_1 = \lambda^*\mathfrak{q}_2^H\mathfrak{q}_1 + \mu^*\mathfrak{q}_3^H\mathfrak{q}_1 = \lambda^* \langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_\mathsf{r} + \mu^* \langle \mathfrak{q}_1, \mathfrak{q}_3 \rangle_\mathsf{r}$.

L'application considérée est donc définie positive et *sesquilinéaire à droite*.

□

Remarque

Cependant, nous remarquons que cette application n'est pas linéaire à gauche par rapport au 1er argument :
$\langle \lambda\mathfrak{q}_1 + \mu\mathfrak{q}_2, \mathfrak{q}_3 \rangle_\mathsf{r} = \mathfrak{q}_3^H\lambda\mathfrak{q}_1 + \mathfrak{q}_3^H\mu\mathfrak{q}_2 \neq \lambda\mathfrak{q}_3^H\mathfrak{q}_1 + \mu\mathfrak{q}_3^H\mathfrak{q}_2 = \lambda \langle \mathfrak{q}_1, \mathfrak{q}_3 \rangle_\mathsf{r} + \mu \langle \mathfrak{q}_2, \mathfrak{q}_3 \rangle_\mathsf{r}$.

Elle n'est pas non plus semi-linéaire à gauche par rapport au 2ième argument :
$\langle \mathfrak{q}_1, \lambda\mathfrak{q}_2 + \mu\mathfrak{q}_3 \rangle_\mathsf{r} = (\lambda\mathfrak{q}_2)\,{}^H\mathfrak{q}_1 + (\mu\mathfrak{q}_3)\,{}^H\mathfrak{q}_1 = \mathfrak{q}_2^H\lambda^*\mathfrak{q}_1 + \mathfrak{q}_3^H\mu^*\mathfrak{q}_1 \neq \lambda^* \langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_\mathsf{r} + \mu^* \langle \mathfrak{q}_1, \mathfrak{q}_3 \rangle_\mathsf{r}$.

Ainsi, l'application proposée n'est pas *sesquilinéaire à gauche*, et donc n'est pas sesquilinéaire. Cependant, ces deux derniers points ne sont pas pénalisants, puisque qu'il n'y a jamais de scalaire multiplié à gauche dans le modèle considéré.

Les considérations ci-dessus peuvent être faites de manière analogue pour le modèle multiplicatif à gauche et l'application linéaire :

$$\begin{aligned} \mathbb{H}^N \times \mathbb{H}^N &\to \mathbb{H} \\ (\mathfrak{q}_1, \mathfrak{q}_2) &\mapsto \langle \mathfrak{q}_1, \mathfrak{q}_2 \rangle_\mathsf{l} = \mathfrak{q}_1\mathfrak{q}_2^H \,, \end{aligned}$$

qui est définie positive et *sesquilinéaire à gauche*.

7.5 Etape de sélection pour le Q-OMPr

Pour le Q-OMPr, la fonction de coût MSE à valeurs réelles est $J = \|\epsilon\|_2^2 = \epsilon^H\epsilon$, avec ϵ défini comme $\epsilon = \epsilon^{k-1} = \phi x + \epsilon^k$. La dérivée de J par rapport à x est calculée ci-dessous.

Remarquons que cette dérivation quaternionique va être effectuée comme la somme des gradients composantes à composantes. Cette manière est appelée pseudo-gradient par Mandic *et al.* qui proposent un opérateur gradient quaternionique dans [MJT11] afin d'étendre aux quaternions l'opérateur gradient complexe [Bra83]. En utilisant leurs nouvelles règles de dérivation, nous obtenons $\partial J/\partial x^* = \phi^H\epsilon - 1/2\,\epsilon^H\phi$. Cependant, maximiser cette expression ne donne pas l'atome optimal. Nous avons eu l'occasion de l'observer sur des données simulées : cette fonctionnelle ne permet pas de retrouver des atomes connus dans des signaux quaternioniques simulés.

$$\begin{aligned} \frac{\partial J}{\partial x} &= \frac{\partial J}{\partial x_a} + \frac{\partial J}{\partial x_b}\mathsf{i} + \frac{\partial J}{\partial x_c}\mathsf{j} + \frac{\partial J}{\partial x_d}\mathsf{k} \\ &= \frac{\partial \epsilon^H}{\partial x_a}\epsilon + \epsilon^H\frac{\partial \epsilon}{\partial x_a} + \frac{\partial \epsilon^H}{\partial x_b}\epsilon\mathsf{i} + \epsilon^H\frac{\partial \epsilon}{\partial x_b}\mathsf{i} + \frac{\partial \epsilon^H}{\partial x_c}\epsilon\mathsf{j} + \epsilon^H\frac{\partial \epsilon}{\partial x_c}\mathsf{j} + \frac{\partial \epsilon^H}{\partial x_d}\epsilon\mathsf{k} + \epsilon^H\frac{\partial \epsilon}{\partial x_d}\mathsf{k}\,. \end{aligned} \tag{7.3}$$

En développant tous les termes de $\epsilon = \phi x + \epsilon^k$ et $\epsilon^H = x^* \phi^H + \epsilon^{k^H}$, nous obtenons :

$$\begin{aligned}
\partial\epsilon/\partial x_a &= \phi_a + \phi_b \mathsf{i} + \phi_c \mathsf{j} + \phi_d \mathsf{k} = \phi & \partial\epsilon^H/\partial x_a &= \phi_a^T - \phi_b^T \mathsf{i} - \phi_c^T \mathsf{j} - \phi_d^T \mathsf{k} = \phi^H \\
\partial\epsilon/\partial x_b &= -\phi_b + \phi_a \mathsf{i} + \phi_d \mathsf{j} - \phi_c \mathsf{k} & \partial\epsilon^H/\partial x_b &= -\phi_b^T - \phi_a^T \mathsf{i} - \phi_d^T \mathsf{j} + \phi_c^T \mathsf{k} \\
\partial\epsilon/\partial x_c &= -\phi_c - \phi_d \mathsf{i} + \phi_a \mathsf{j} + \phi_b \mathsf{k} & \partial\epsilon^H/\partial x_c &= -\phi_c^T + \phi_d^T \mathsf{i} - \phi_a^T \mathsf{j} - \phi_b^T \mathsf{k} \\
\partial\epsilon/\partial x_d &= -\phi_d + \phi_c \mathsf{i} - \phi_b \mathsf{j} + \phi_a \mathsf{k} & \partial\epsilon^H/\partial x_d &= -\phi_d^T - \phi_c^T \mathsf{i} + \phi_b^T \mathsf{j} - \phi_a^T \mathsf{k} \,.
\end{aligned}$$

En remplaçant ces huit termes dans l'Eq. (7.3), nous avons :

$$\begin{aligned}
\frac{\partial J}{\partial x} &= \phi^H \epsilon + \epsilon^H \phi \\
&\quad + (-\phi_b^T - \phi_a^T \mathsf{i} - \phi_d^T \mathsf{j} + \phi_c^T \mathsf{k})(-\epsilon_b + \epsilon_a \mathsf{i} + \epsilon_d \mathsf{j} - \epsilon_c \mathsf{k}) && (7.4) \\
&\quad + \epsilon^H (-\phi_a - \phi_b \mathsf{i} - \phi_c \mathsf{j} - \phi_d \mathsf{k}) \\
&\quad + (-\phi_c^T + \phi_d^T \mathsf{i} - \phi_a^T \mathsf{j} - \phi_b^T \mathsf{k})(-\epsilon_c - \epsilon_d \mathsf{i} + \epsilon_a \mathsf{j} + \epsilon_b \mathsf{k}) && (7.5) \\
&\quad + \epsilon^H (-\phi_a - \phi_b \mathsf{i} - \phi_c \mathsf{j} - \phi_d \mathsf{k}) \\
&\quad + (-\phi_d^T - \phi_c^T \mathsf{i} + \phi_b^T \mathsf{j} - \phi_a^T \mathsf{k})(-\epsilon_d + \epsilon_c \mathsf{i} - \epsilon_b \mathsf{j} + \epsilon_a \mathsf{k}) && (7.6) \\
&\quad + \epsilon^H (-\phi_a - \phi_b \mathsf{i} - \phi_c \mathsf{j} - \phi_d \mathsf{k}) \\
&= \phi^H \epsilon - 2\epsilon^H \phi + (7.4) + (7.5) + (7.6) \,. && (7.7)
\end{aligned}$$

En développant les trois termes (7.4), (7.5) et (7.6), en additionnant et en factorisant, nous obtenons :

$$(7.4) + (7.5) + (7.6) = \phi^H \epsilon + 2\epsilon^H \phi \,. \tag{7.8}$$

Avec l'Eq. (7.7), nous avons finalement :

$$\begin{aligned}
\frac{\partial J}{\partial x} &= \phi^H \epsilon - 2\epsilon^H \phi + \phi^H \epsilon + 2\epsilon^H \phi \\
&= 2\phi^H \epsilon = 2 \left\langle \epsilon, \phi \right\rangle_{\mathsf{r}} \,. && (7.9)
\end{aligned}$$

Ainsi, nous pouvons conclure que l'atome qui produit la plus forte décroissance (en valeur absolue) de la MSE $\|\epsilon\|_2^2$ est le plus corrélé au résidu, comme dans le cas complexe.

7.6 Etape de sélection pour le Q-OMPl

Pour le Q-OMPl, la fonction de coût MSE est $J = \|\epsilon\|_2^2 = \epsilon\epsilon^H$, avec ϵ défini comme $\epsilon = \epsilon^{k-1} = x\phi + \epsilon^k$. La dérivée de J par rapport à x est calculée ci-dessous.

$$\begin{aligned}
\frac{\partial J}{\partial x} &= \frac{\partial J}{\partial x_a} + \frac{\partial J}{\partial x_b}\mathsf{i} + \frac{\partial J}{\partial x_c}\mathsf{j} + \frac{\partial J}{\partial x_d}\mathsf{k} \\
&= \frac{\partial \epsilon}{\partial x_a}\epsilon^H + \epsilon\frac{\partial \epsilon^H}{\partial x_a} + \frac{\partial \epsilon}{\partial x_b}\epsilon^H\mathsf{i} + \epsilon\frac{\partial \epsilon^H}{\partial x_b}\mathsf{i} + \frac{\partial \epsilon}{\partial x_c}\epsilon^H\mathsf{j} + \epsilon\frac{\partial \epsilon^H}{\partial x_c}\mathsf{j} + \frac{\partial \epsilon}{\partial x_d}\epsilon^H\mathsf{k} + \epsilon\frac{\partial \epsilon^H}{\partial x_d}\mathsf{k} \,. && (7.10)
\end{aligned}$$

En développant tous les termes de $\epsilon = x\phi + \epsilon^k$ et $\epsilon^H = \phi^H x^* + \epsilon^{k^H}$, les huit termes de l'Eq. (7.10) deviennent :

$$\begin{aligned}
\frac{\partial J}{\partial x} &= \phi\epsilon^H + \epsilon\phi^H \\
&\quad +(-\phi_b + \phi_a\mathsf{i} - \phi_d\mathsf{j} + \phi_c\mathsf{k})(\epsilon_b^T + \epsilon_a^T\mathsf{i} - \epsilon_d^T\mathsf{j} + \epsilon_c^T\mathsf{k}) && (7.11)\\
&\quad +\epsilon(\phi_a^T - \phi_b^T\mathsf{i} - \phi_c^T\mathsf{j} - \phi_d^T\mathsf{k}) \\
&\quad +(-\phi_c + \phi_d\mathsf{i} + \phi_a\mathsf{j} - \phi_b\mathsf{k})(\epsilon_c^T + \epsilon_d^T\mathsf{i} + \epsilon_a^T\mathsf{j} - \epsilon_b^T\mathsf{k}) && (7.12)\\
&\quad +\epsilon(\phi_a^T - \phi_b^T\mathsf{i} - \phi_c^T\mathsf{j} - \phi_d^T\mathsf{k}) \\
&\quad +(-\phi_d - \phi_c\mathsf{i} + \phi_b\mathsf{j} + \phi_a\mathsf{k})(\epsilon_d^T - \epsilon_c^T\mathsf{i} + \epsilon_b^T\mathsf{j} + \epsilon_a^T\mathsf{k}) && (7.13)\\
&\quad +\epsilon(\phi_a^T - \phi_b^T\mathsf{i} - \phi_c^T\mathsf{j} - \phi_d^T\mathsf{k}) \\
&= \phi\epsilon^H + \epsilon\phi^H + (7.11) + \epsilon\phi^H + (7.12) + \epsilon\phi^H + (7.13) + \epsilon\phi^H \\
&= \phi\epsilon^H + 4\epsilon\phi^H + (7.11) + (7.12) + (7.13)\,. && (7.14)
\end{aligned}$$

En développant les trois termes (7.11), (7.12) et (7.13), en additionnant et en factorisant, nous obtenons :

$$(7.11) + (7.12) + (7.13) = -\phi\epsilon^H - 2\epsilon\phi^H\,. \tag{7.15}$$

Avec l'Eq. (7.14), nous avons finalement :

$$\begin{aligned}
\frac{\partial J}{\partial x} &= \phi\epsilon^H + 4\epsilon\phi^H - \phi\epsilon^H - 2\epsilon\phi^H \\
&= 2\epsilon\phi^H = 2\left\langle \epsilon, \phi \right\rangle_{\mathsf{I}}\,. && (7.16)
\end{aligned}$$

De la même manière, nous pouvons conclure que l'atome qui produit la plus forte décroissance (en valeur absolue) de la MSE $\|\epsilon\|_2^2$ est le plus corrélé au résidu. Comme pour le Q-OMPr, l'opérateur gradient quaternionique [MJT11], qui donne $\partial J/\partial x^* = \epsilon\phi^H - 1/2\ \phi\epsilon^H$, ne permet pas de sélectionner l'atome optimal.

7.7 Problème de Procrustes orthogonal

Le problème du recalage de Procrustes orthogonal s'écrit ainsi :

$$\min_{x,R} \| \boldsymbol{y} - x\, R\, \boldsymbol{\phi} \|_F^2 \quad \text{t.q.} \quad RR^T = Id\,. \tag{7.17}$$

Nous développons la norme de Frobenius :

$$\begin{aligned}
\| \boldsymbol{y} - x\, R\, \boldsymbol{\phi} \|_F^2 &= \mathrm{Tr}(\boldsymbol{y}\boldsymbol{y}^T) - 2x\mathrm{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) + x^2\mathrm{Tr}(R\boldsymbol{\phi}\boldsymbol{\phi}^T R^T)\,, && (7.18)\\
&= \|\boldsymbol{y}\|_F^2 - 2x\mathrm{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) + x^2\mathrm{Tr}(\boldsymbol{\phi}\boldsymbol{\phi}^T R^T R) \qquad \text{car } \mathrm{Tr}(AB) = \mathrm{Tr}(BA)\,, && (7.19)\\
&= \|\boldsymbol{y}\|_F^2 - 2x\mathrm{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) + x^2 \qquad \text{car } R^T R = Id \text{ et car } \boldsymbol{\phi} \text{ est normé.} && (7.20)
\end{aligned}$$

Le coefficient x et la matrice de rotation R peuvent donc être estimés indépendamment.

Estimation de R

Cette démonstration est inspirée de [Hig95]. Suite au calcul ci-dessus, la minimisation de l'Eq. (7.17) est finalement équivalente à : $\max_R \text{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T)$.

$$\begin{aligned}
\text{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) &= \text{Tr}(M_c R^T) \qquad \text{en posant } M_c = \boldsymbol{y}\boldsymbol{\phi}^T\,, && (7.21)\\
&= \text{Tr}(U\Sigma_1 V^T R^T) \qquad \text{par SVD de } M_c\,, && (7.22)\\
&= \text{Tr}(\Sigma_1 V^T R^T U) && (7.23)
\end{aligned}$$

or $Q = V^T R^T U \in \mathbb{R}^{V\times V}$ est une matrice orthogonale, nous avons donc $\forall v \in \mathbb{N}_V, |q_{vv}| \leq 1$ et

$$= \text{Tr}(\Sigma_1 Q) = \sum_{v=1}^{V} \sigma_v\ q_{vv} \leq \sum_{v=1}^{V} \sigma_v. \qquad (7.24)$$

L'égalité est vérifiée pour $Q = Id$, soit $R = UV^T$.

Nous ne détaillons pas ici la démonstration de [Ume91] et [Kan94] qui introduisent la matrice $\Sigma_2 = \text{diag}(1, 1, \det(UV^T))$ et qui définissent $R = U\Sigma_2 V^T$, assurant ainsi que R est une rotation ($\det R = 1$) et non une réflexion ($\det R = -1$).

Estimation de x

En utilisant l'Eq. (7.20), nous dérivons le critère par rapport à x :

$$\frac{\partial\left(\|\,\boldsymbol{y} - x\,R\,\boldsymbol{\phi}\,\|_F^2\right)}{\partial x} = \frac{\partial\left(\|\boldsymbol{y}\|_F^2 - 2x\text{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) + x^2\right)}{\partial x} = -2\text{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) + 2x\,. \qquad (7.25)$$

En annulant la dérivée pour obtenir l'optimum, nous avons :

$$\begin{aligned}
x &= \text{Tr}(\boldsymbol{y}\boldsymbol{\phi}^T R^T) = \text{Tr}(R\boldsymbol{\phi}\boldsymbol{y}^T) && (7.26)\\
&= \text{Tr}(U\Sigma_2 V^T (U\Sigma_1 V^T)^T) = \text{Tr}(\Sigma_2\Sigma_1) \geq 0\,. && (7.27)
\end{aligned}$$

Bibliographie

[Bra83] D.H. Brandwood : A complex gradient operator and its application in adaptive array theory. *IEE Proceedings F - Communications, Radar and Signal Processing*, 130:11–16, 1983.

[Hig95] N. Higham : Matrix procrustes problems. Rapport technique, University of Manchester, 1995.

[Kan94] K. Kanatani : Analysis of 3-D rotation fitting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16:543–549, 1994.

[MJT11] D.P. Mandic, C. Jahanchahi et C.C. Took : A quaternion gradient operator and its applications. *IEEE Signal Processing Letters*, 18:47–50, 2011.

[Ume91] S. Umeyama : Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:376–380, 1991.

Bibliographie générale

[AC12] B. Augereau et P. Carré : Hypercomplex polynomial wavelet packet application for color image. *In Conf. Applied Geometric Algebras in Computer Science and Engineering AGACSE*, 2012.

[AE08] M. Aharon et M. Elad : Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM J. Imaging Sciences*, 1:228–247, 2008.

[AEB06a] M. Aharon, M. Elad et A.M. Bruckstein : K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54:4311–4322, 2006.

[AEB06b] M. Aharon, M. Elad et A.M. Bruckstein : On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Linear Algebra and its Applications*, 416:48–67, 2006.

[Aha06] M. Aharon : *Overcomplete Dictionaries for Sparse Representation of Signals*. Thèse de doctorat, Technion - Israel Institute of Technology, 2006.

[AHB87] K.S. Arun, T.S. Huang et S.D. Blostein : Least-squares fitting of two 3-D point sets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-9:698–700, 1987.

[AHSE01] S.O. Aase, J.H. Husøy, K. Skretting et K. Engan : Optimized signal expansions for sparse representation. *IEEE Trans. on Signal Processing*, 49:1087–1096, 2001.

[ASKK11] I. Akhter, Y. Sheikh, S. Khan et T. Kanade : Trajectory space : A dual representation for nonrigid structure from motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33: 1442–1456, 2011.

[ASS98] C.W. Anderson, E.A. Stolz et S. Shamsunder : Multivariate autoregressive models for classification of spontaneous electroencephalographic signals during mental tasks. *IEEE Trans. on Biomedical Engineering*, 45:277–286, 1998.

[BB08] L. Bottou et O. Bousquet : The tradeoffs of large scale learning. *In Advances in Neural Information Processing Systems NIPS*, volume 20, pages 161–168, 2008.

[BBC11] S. Becker, J. Bobin et E.J. Candès : NESTA : A fast and accurate first-order method for sparse recovery. *SIAM J. Imaging Sciences*, 4:1–39, 2011.

[BD06] T. Blumensath et M.E. Davies : Sparse and shift-invariant representations of music. *IEEE Trans. on Audio, Speech, and Language Processing*, 14:50–57, 2006.

[BD07] T. Blumensath et M.E. Davies : On the difference between orthogonal matching pursuit and orthogonal least squares. Rapport technique, Edinburgh University, 2007.

[BD08] T. Blumensath et M.E. Davies : Gradient pursuits. *IEEE Trans. on Signal Processing*, 56:2370–2382, 2008.

[BD09] T. Blumensath et M.E. Davies : Iterative hard thresholding for compressed sensing. *Appl. Comput. Harmon. Anal.*, 27:265–274, 2009.

[BD10] T. Blumensath et M.E. Davies : Normalized iterative hard thresholding : Guaranteed stability and performance. *IEEE Journal of Selected Topics in Signal Processing*, 4:298–309, 2010.

[BDE09] A.M. Bruckstein, D.L. Donoho et M. Elad : From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51:34–81, 2009.

[BDF07] J.M. Bioucas-Dias et M.A.T. Figueiredo : A new TwIST : Two-step iterative shrinkage/ thresholding algorithms for image restoration. *IEEE Trans. on Image Processing*, 16:2992–3004, 2007.

[BDS+05] D. Baron, M.F. Duarte, S. Sarvotham, M.B. Wakin et R.G. Baraniuk : An information-theoretic approach to distributed compressed sensing. *In Proc. Allerton Conf. Communication, Control, and Computing*, 2005.

[BE02] A.M. Bruckstein et M. Elad : A generalized uncertainty principle and sparse representation in pairs of bases. *IEEE Trans. on Information Theory*, 48:2558–2567, 2002.

[BHB00] C. Bregler, A. Hertzmann et H. Biermann : Recovering non-rigid 3D shape from image streams. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR*, pages 690–696, 2000.

[BLM12] Q. Barthélemy, A. Larue et J.I. Mars : 3D rotation invariant decomposition of motion signals. *In Computer Vision – ECCV 2012*, volume 7585 de *Lecture Notes in Computer Science*, pages 172–182, 2012.

[Blu05] T. Blumensath : *Bayesian Modelling of Music : Algorithmic Advances and Experimental Studies of Shift-Invariant Sparse Coding*. Thèse de doctorat, University of London, 2005.

[BM92] P.J. Besl et H.D. McKay : A method for registration of 3-D shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14:239–256, 1992.

[BM13] J. Bruna et S. Mallat : Invariant scattering convolution networks. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35:1872–1886, 2013.

[BMB+09] J.P. Bandera, R. Marfil, A. Bandera, J.A. Rodriguez, L. Molina-Tanco et F. Sandoval : Fast gesture recognition based on a two-level representation. *Pattern Recognition Letters*, 30:1181–1189, 2009.

[Boi74] N. Boileau : *L'Art politique*. 1674.

[BPTC09] C.G. Bénar, T. Papadopoulo, B. Torrésani et M. Clerc : Consensus matching pursuit for multi-trial EEG signals. *J. Neuroscience Methods*, 180:161–170, 2009.

[Bra83] D.H. Brandwood : A complex gradient operator and its application in adaptive array theory. *IEE Proceedings F - Communications, Radar and Signal Processing*, 130:11–16, 1983.

[BS98] R. Benjemaa et F. Schmitt : A solution for the registration of multiple 3D point sets using unit quaternions. *In Proc. Eur. Conf. Computer Vision ECCV '98*, pages 34–50, 1998.

[BS02] B. Blankertz et G. Schalk : 2nd Wadsworth BCI Dataset (P300 Evoked Potentials), 2002.

[BSGL96] R. Bergevin, M. Soucy, H. Gagnon et D. Laurendeau : Towards a general multi-view registration technique. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:540–547, 1996.

[BSSS86] C.M. Bastuscheck, E. Schonberg, J.T. Schwartz et M. Sharir : Object recognition by 3-Dimensional curve matching. *Int. Journal of Intelligent Systems*, 1:105–132, 1986.

[BT09] A. Beck et M. Teboulle : A fast iterative shrinkage-threshold algorithm for linear inverse problems. *SIAM J. Imaging Sciences*, 2:183–202, 2009.

[BTL+08] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe et K.R. Müller : Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, 41:581–607, 2008.

[CARKD99] S.F. Cotter, R. Adler, R.D. Rao et K. Kreutz-Delgado : Forward sequential algorithms for best basis selection. *IEE Proceedings - Vision, Image and Signal Processing*, 146:235–244, 1999.

[CAS05] A. Croitoru, P. Agouris et A. Stefanidis : 3D trajectory matching by pose normalization. *In Proc. ACM Int. Workshop on Geographic Information Systems GIS '05*, pages 153–162, 2005.

[CBA13] S. Chevallier, Q. Barthélemy et J. Atif : Metrics for multivariate dictionaries. Preprint arXiv :1302.4242, 2013.

[CBL89] S. Chen, S.A. Billings et W. Luo : Orthogonal least squares methods and their application to non-linear system identification. *Int. Journal of Control*, 50:1873–1896, 1989.

[CD99] E.J. Candès et D. Donoho : Ridgelets : The key to higher-dimensional intermittency ? *Phil. Trans. R. Soc. Lond. A.*, 357:2495–2509, 1999.

[CD00] E.J. Candès et D.L. Donoho : *Curvelets - A Surprisingly Effective Nonadaptive Representation For Objects with Edges*, pages 105 – 120. Curves and Surfaces, Vanderbilt University Press, 2000.

[CDS98] S. Chen, D.L. Donoho et M.A. Saunders : Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20:33–61, 1998.

[CGPJ08] M. Congedo, C. Gouy-Pailler et C. Jutten : On the blind source separation of human electroencephalogram by approximate joint diagonalization of second order statistics. *Clinical Neurophysiology*, 119:2677–2686, 2008.

[CH06] J. Chen et X. Huo : Theoretical results on sparse representations of Multiple-Measurement Vectors. *IEEE Trans. on Signal Processing*, 54:4634–4643, 2006.

[CJ10] P. Comon et C. Jutten : *Handbook of Blind Source Separation, Independent Component Analysis and Applications*. New York : Academic, 2010.

[Com94] P. Comon : Independent Component Analysis, a new concept ? *Signal Process.*, 36:287–314, 1994.

[CR05] E. Candès et J. Romberg : ℓ_1-MAGIC : Recovery of sparse signals via convex programming. Rapport technique, California Inst. Technol., 2005.

[CREKD05] S.F. Cotter, B.D. Rao, K. Engan et K. Kreutz-Delgado : Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE Trans. on Signal Processing*, 53:2477–2488, 2005.

[CT05] E.J. Candès et T. Tao : Decoding by linear programming. *IEEE Trans. on Information Theory*, 51:4203–4215, 2005.

[CW95] S. Chen et J. Wigger : Fast orthogonal least squares algorithm for efficient subset model selection. *IEEE Trans. on Signal Processing*, 43:1713–1715, 1995.

[CW05] P.L. Combettes et V.R. Wajs : Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4:1168–1200, 2005.

[CWB08] E. Candès, M.B. Wakin et S.P. Boyd : Enhancing sparsity by reweighted ℓ_1 minimization. *J. Fourier Anal. Appl.*, 14:877–905, 2008.

[Dau88] I. Daubechies : Time-frequency localization operators : a geometric phase space approach. *IEEE Trans. on Information Theory*, 34:605–612, 1988.

[Dav94] G. Davis : *Adaptive Nonlinear Approximations.* Thèse de doctorat, New York University, 1994.

[DB01] P.J. Durka et K.J. Blinowska : A unified time-frequency parametrization of EEGs. *IEEE Engineering in Medicine and Biology Magazine*, 20:47–53, 2001.

[DDDM04] I. Daubechies, M. Defrise et C. De Mol : An iterative algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.*, LVII:1413–1457, 2004.

[DET06] D.L. Donoho, M. Elad et V. Temlyakov : Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. on Information Theory*, 52:6–18, 2006.

[DF12] O. Dikmen et C. Févotte : Maximum marginal likelihood estimation for nonnegative dictionary learning in the Gamma-Poisson model. *IEEE Trans. on Signal Processing*, 60:5163–5175, 2012.

[DH01] D.L. Donoho et X. Huo : Uncertainty principle and ideal atomic decompositions. *IEEE Trans. on Information Theory*, 47:2845–2862, 2001.

[DHS01] R.O. Duda, P.E. Hart et D.G. Stork : *Pattern Classification.* 2nd edition, Wiley-Interscience, 2001.

[DL12] Z. Dai et J. Lücke : Unsupervised learning of translation invariant occlusive components. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR*, pages 2400–2407, 2012.

[DM09] W. Dai et O. Milenkovic : Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. on Information Theory*, 55:2230–2249, 2009.

[DMMM+05] P.J. Durka, A. Matysiak, E. Martínez-Montes, P.A. Valdés-Sosa et K.J. Blinowska : Multichannel matching pursuit and EEG inverse solutions. *J. Neuroscience Methods*, 148:49–59, 2005.

[DMZ94] G. Davis, S. Mallat et Z. Zhang : Adaptive time-frequency decompositions with matching pursuits. *Optical Engineering*, 33:2183–2191, 1994.

[DSAS11] C. D'Avanzo, S. Schiff, P. Amodio et G. Sparacino : A Bayesian method to estimate single-trial event-related potentials with application to the study of the P300 variability. *J. Neuroscience Methods*, 198:114–124, 2011.

[DT08] D.L. Donoho et Y. Tsaig : Fast solution of ℓ_1-norm minimization problems when the solution may be sparse. *IEEE Trans. on Information Theory*, 54:4789–4812, 2008.

[DTDS06] D.L. Donoho, Y. Tsaig, I. Drori et J.L. Starck : Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. Rapport technique, Stanford University, 2006.

[Dur07] P.J. Durka : *Matching Pursuit and Unification in EEG Analysis.* Artech House, 2007.

[DVT96] R.A. De Vore et V.N. Temlyakov : Some remarks on greedy algorithms. *Advances in Computational Mathematics*, 5:173–187, 1996.

[EA06] M. Elad et M. Aharon : Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. on Image Processing*, 15:3736–3745, 2006.

[EAH00] K. Engan, S.O. Aase et J.H. Husøy : Multi-frame compression : theory and design. *Signal Process.*, 80:2121–2140, 2000.

[EHJT04] B. Efron, T. Hastie, I. Johnstone et R. Tibshirani : Least Angle Regression. *Ann. Stat.*, 32:407–499, 2004.

[Ela06] M. Elad : Why simple shrinkage is still relevant for redundant representations. *IEEE Trans. on Information Theory*, 52:5559–5569, 2006.

[ELF97] D.W. Eggert, A. Lorusso et R.B. Fisher : Estimating 3-D rigid body transformations : a comparison of four major algorithms. *Machine Vision and Applications*, 9:272–290, 1997.

[ER10] Y.C. Eldar et H. Rauhut : Average case analysis of multichannel sparse recovery using convex relaxation. *IEEE Trans. on Information Theory*, 56:505–519, 2010.

[ES07] T.A. Ell et S.J. Sangwine : Hypercomplex Fourier transforms of color images. *IEEE Trans. on Image Processing*, 16:22–35, 2007.

[ESH07] K. Engan, K. Skretting et J.H. Husøy : Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation. *Digit. Signal Process.*, 17:32–49, 2007.

[FA10] A. Frank et A. Asuncion : UCI machine learning repository, 2010.

[FNW07] M.A.T. Figueiredo, R.D. Nowak et S.J. Wright : Gradient projection for sparse reconstruction : Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics in Signal Processing*, 1:586–597, 2007.

[FSBM10] M.J. Fadili, J.-L. Starck, J. Bobin et Y. Moudden : Image decomposition and separation using sparse representations : An overview. *Proceedings of the IEEE*, 98:983–994, 2010.

[Fuc04] J.-J. Fuchs : On sparse representations in arbitrary redundant bases. *IEEE Trans. on Information Theory*, 50:1341–1344, 2004.

[GBB+05] G. Gibert, G. Bailly, D. Beautemps, F. Elisei et R. Brun : Analysis and synthesis of the three-dimensional movements of the head, face, and hand of a speaker using cued speech. *Journal of the Acoustical Society of America*, 118:1144–1153, 2005.

[GD04] J.C. Gower et G.B. Dijksterhuis : *Procrustes Problems*. Oxford Statistical Science Series, 30, 2004.

[GHANZ07] M. Gratkowski, J. Haueisen, L. Arendt-Nielsen et F. Zanow : Topographic matching pursuit of spatio-temporal bioelectromagnetic data. *Przeglad Elektrotechniczny*, 83:138–141, 2007.

[GHANZ08] M. Gratkowski, J. Haueisen, A. Arendt-Nielsen, L. Chen et F. Zanow : Decomposition of biomedical signals in spatial and time-frequency modes. *Methods of Information in Medicine*, 47:26–37, 2008.

[Gib06] G. Gibert : *Conception et évaluation d'un système de synthèse 3D de Langue française Parlï£¡e Complétée (LPC) à partir du texte*. Thèse de doctorat, Institut National Polytechnique de Grenoble, 2006.

[GN03] R. Gribonval et M. Nielsen : Sparse representations in unions of bases. *IEEE Trans. on Information Theory*, 49:3320–3325, 2003.

[GN05] R. Gribonval et M. Nielsen : Beyond sparsity : Recovering structured representations by ℓ_1-minimization and greedy algorithms. -Application to the analysis of sparse underdetermined ICA-. Rapport technique PI-1684, IRISA, 2005.

[Gow75] J.C. Gower : Generalized Procrustes analysis. *Psychometrika*, 40:33–51, 1975.

[Gow10] J.C. Gower : Procrustes methods. *Wiley Interdisciplinary Reviews : Computational Statistics*, 2:503–508, 2010.

[GPCB+10] C. Gouy-Pailler, M. Congedo, C. Brunner, C. Jutten et G. Pfurtscheller : Nonstationary brain source separation for multiclass motor imagery. *IEEE Trans. on Biomedical Engineering*, 57:469–478, 2010.

[GR97] I.F. Gorodnitsky et B. Rao : Sparse signal reconstruction from limited data using FOCUSS : A re-weighted minimum norm algorithm. *IEEE Trans. on Signal Processing*, 45:600–616, 1997.

[Gri02] R. Gribonval : Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '02*, pages 3057–3060, 2002.

[Gri03] R. Gribonval : Piecewise linear source separation. *In Proc. SPIE 5207*, pages 297–310, 2003.

[GRKN07] H. Grosse, R. Raina, R. Kwong et A.Y. Ng : Shift-invariant sparse coding for audio classification. *In Proc. Conf. Uncertainty in Artificial Intelligence UAI*, pages 149–158, 2007.

[GRSV07] R. Gribonval, H. Rauhut, K. Schnass et P. Vandergheynst : Atoms of all channels, unite ! Average case analysis of multi-channel sparse recovery using greedy algorithms. Rapport technique PI-1848, IRISA, 2007.

[GS10] R. Gribonval et K. Schnass : Dictionary identification — sparse matrix-factorization via ℓ_1-minimization. *IEEE Trans. on Information Theory*, 56:3523–3539, 2010.

[GT10] B.V. Gowreesunker et A.H. Tewfik : Learning sparse representation using iterative subspace identification. *IEEE Trans. on Signal Processing*, 58:3055–3065, 2010.

[GWW11] Q. Geng, H. Wang et J. Wright : On the local correctness of ℓ_1 minimization for dictionary learning. *Preprint arXiv :1101.5672*, 2011.

[HA06] K. Huang et S. Aviyente : Sparse representation for signal classification. *In Advances in Neural Information Processing Systems NIPS*, pages 609–616, 2006.

[Han06] A.J. Hanson : *Visualizing quaternions*. Morgan-Kaufmann/Elsevier, 2006.

[HBRN10] J. Haupt, W.U. Bajwa, G. Raz et R. Nowak : Toeplitz compressed sensing matrices with applications to sparse channel estimation. *IEEE Trans. on Information Theory*, 56:5862–5875, 2010.

[HCdRM11] B. Hamner, R. Chavarriaga et J. del R. Millán : Learning dictionaries of spatial and temporal EEG primitives for brain-computer interfaces. *In Workshop on structured sparsity : learning and inference ; ICML 2011*, 2011.

[Hig95] N. Higham : Matrix procrustes problems. Rapport technique, University of Manchester, 1995.

[HO00] A. Hyvärinen et E. Oja : Independent component analysis : algorithms and applications. *Neural Networks*, 13:411–430, 2000.

[Hor87] B.K.P. Horn : Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4:629–642, 1987.

[Hor88] B.K.P. Horn : Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 4:1127–1135, 1988.

[HTF09] T. Hastie, R. Tibshirani et J. Friedman : *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. Springer New York Inc., 2009.

[HYHJ+12] P.-S. Huang, J. Yang, M. Hasegawa-Johnson, F. Liang et T.S. Huang : Pooling robust shift-invariant sparse representations of acoustic signals. *In Conf. Int. Speech Communication Association, Interspeech 2012*, 2012.

[HYK+01] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi et K. Tanie : Planning walking patterns for a biped robot. *IEEE Trans. on Robotics and Automation*, 17:280–289, 2001.

[HYZ08] E.T. Hale, W. Yin et Y. Zhang : A fixed-point continuation method for ℓ_1-minimization : Methodology and convergence. *SIAM J. Optim.*, 19:1107–1130, 2008.

[JCP+11] N. Jrad, M. Congedo, R. Phlypo, S. Rousseau, R. Flamary, F. Yger et A. Rakotomamonjy : sw-SVM : sensor weighting support vector machines for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 8:1–11, 2011.

[JDCP11] L. Jacques, L. Duval, C. Chaux et G. Peyré : A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity. *Signal Process.*, 91:2699–2730, 2011.

[JGB12] R. Jenatton, R. Gribonval et F. Bach : Local stability and robustness of sparse dictionary learning in the presence of noise. *Preprint arXiv :1210.0685*, 2012.

[JLD11] Z. Jiang, Z. Lin et L.S. Davis : Learning a discriminative dictionary for sparse coding via label consistent K-SVD. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2011*, pages 1697–1704, 2011.

[JMH+00] T.-P. Jung, S. Makeig, C. Humphries, T.-W. Lee, M.J. McKeown, V. Iragui et T.J. Sejnowski : Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, 37:163–178, 2000.

[JSM+11] M. Jörn, C. Sielużycki, M.A. Matysiak, J. Żygierewicz, H. Scheich, P.J. Durka et R. König : Single-trial reconstruction of auditory evoked magnetic fields by means of template matching pursuit. *J. Neuroscience Methods*, 199:119–128, 2011.

[JTJ+11] S. Javidi, C.C. Took, C. Jahanchahi, N. Le Bihan et D.P. Mandic : Blind extraction of improper quaternion sources. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '11*, pages 3708–3711, 2011.

[JV99] P. Jaśkowski et R. Verleger : Amplitudes and latencies of single-trial ERP's estimated by a maximum-likelihood method. *IEEE Trans. on Biomedical Engineering*, 46:987–993, 1999.

[JVLG05] P. Jost, P. Vandergheynst, S. Lesage et R. Gribonval : Learning redundant dictionaries with translation invariance property : the MoTIF algorithm. *In Workshop Structure et parcimonie pour la représentation adaptative de signaux SPARS '05*, 2005.

[JVLG06] P. Jost, P. Vandergheynst, S. Lesage et R. Gribonval : MoTIF : An efficient algorithm for learning translation invariant dictionaries. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '06*, pages 857–860, 2006.

[Kan94] K. Kanatani : Analysis of 3-D rotation fitting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16:543–549, 1994.

[KDMR+03] K. Kreutz-Delgado, J.F. Murray, B.D. Rao, K. Engan, T. Lee et T.J. Sejnowski : Dictionary learning algorithms for sparse representation. *Neural Comput.*, 15:349–396, 2003.

[KF08] E. Kokiopoulou et P. Frossard : Semantic coding by supervised dimensionality reduction. *IEEE Trans. on Multimedia*, 10:806–818, 2008.

[KKL+07] S.J. Kim, K. Koh, M. Lustig, S. Boyd et D. Gorinevsky : An interior-point method for large-scale ℓ_1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1:606–617, 2007.

[KMLVS01] T. Koenig, F. Marti-Lopez et P. Valdés-Sosa : Topographic time-frequency decomposition of the EEG. *NeuroImage*, 14:383–390, 2001.

[KPJR91] B. Kamgar-Parsi, J.L. Jones et A. Rosenfeld : Registration of multiple overlapping range images : scenes without distinctive features. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:857–871, 1991.

[KST+06] K.H. Knuth, A.S. Shah, W.A. Truccolo, M. Ding, S.L. Bressler et C.E. Schroeder : Differentially variable component analysis : Identifying multiple evoked components using trial-to-trial variability. *Journal of Neurophysiology*, 95:3257–3276, 2006.

[KSU10] T. Kim, G. Shakhnarovich et R. Urtasun : Sparse coding for learning interpretable spatio-temporal primitives. *In Advances in Neural Information Processing Systems NIPS*, pages 1117–1125, 2010.

[LBBH98] Y. LeCun, L. Bottou, Y. Bengio et P. Haffner : Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

[LBRN07] H. Lee, A. Battle, R. Raina et A.Y. Ng : Efficient sparse coding algorithms. *In Advances in Neural Information Processing Systems NIPS*, pages 801–808, 2007.

[LeC12] Y. LeCun : Learning invariant feature hierarchies. *In Computer Vision – ECCV 2012*, volume 7583 de *Lecture Notes in Computer Science*, pages 496–505, 2012.

[LGBB05] S. Lesage, R. Gribonval, F. Bimbot et L. Benaroya : Learning unions of orthonormal bases with thresholded singular value decomposition. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '05*, pages 293–296, 2005.

[LGP+08] M.-F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli et D. Farina : Multi-channel surface EMG classification using support vector machines and signal-based wavelet optimization. *Biomedical Signal Processing and Control*, 3:169–174, 2008.

[LKG06] S. Lesage, S. Krstulović et R. Gribonval : Under-determined source separation : comparison of two approaches based on sparse decompositions. *In Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation*, pages 633–640, 2006.

[LKP09] R. Li, A. Keil et J.C. Principe : Single-trial P300 estimation with a spatiotemporal filtering method. *J. Neuroscience Methods*, 177:488–496, 2009.

[LO99] M.S. Lewicki et B.A. Olshausen : A probabilistic framework for the adaptation and comparison of image codes. *Journal of the Optical Society of America A*, 16:1587–1601, 1999.

[LPBF09] R. Li, J.C. Principe, M. Bradley et V. Ferrari : A spatiotemporal filtering methodology for single-trial ERP component estimation. *IEEE Trans. on Biomedical Engineering*, 56:83–92, 2009.

[LPI97] D.H. Lange, H. Pratt et G.F. Inbar : Modeling and estimation of single evoked brain potential components. *IEEE Trans. on Biomedical Engineering*, 44:791–799, 1997.

[LS98] M.S. Lewicki et T.J. Sejnowski : Learning overcomplete representations. *Neural Comput.*, 12:337–365, 1998.

[LS99] M.S. Lewicki et T.J. Sejnowski : Coding time-varying signals using sparse, shift-invariant representations. *In Advances in Neural Information Processing Systems II*, pages 730–736, 1999.

[LT03a] D. Leviathan et V.N. Temlyakov : Simultaneous approximation by greedy algorithms. Rapport technique, Univ. of South Carolina at Columbia, 2003.

[LT03b] A. Lutoborski et V.N. Temlyakov : Vector greedy algorithms. *J. Complex.*, 19:458–473, 2003.

[LT05] D. Leviathan et V.N. Temlyakov : Simultaneous greedy approximation in Banach spaces. *J. Complex.*, 21:275–293, 2005.

[Mal09] S. Mallat : *A Wavelet Tour of signal processing.* 3rd edition, New-York : Academic, 2009.

[Mal12] S. Mallat : Group invariant scattering. *Commun. Pure Appl. Math.*, 65:1331–1398, 2012.

[MBM03] D. Melkonian, T.D. Blumenthal et R. Meares : High-resolution fragmentary decomposition - a model-based method of non-stationary electrophysiological signal analysis. *J. Neuroscience Methods*, 131:149–159, 2003.

[MBM+12] A. Mouraud, Q. Barthélemy, A. Mayoue, C. Gouy-Pailler, A. Larue et H. Paugam-Moisy : From neuronal cost-based metrics to sparse coded signals classification. *In Proc. Eur. Symp. Artificial Neural Networks, Computational Intelligence and Machine Learning ESANN*, pages 311–316, 2012.

[MBOL12] A. Mayoue, Q. Barthélemy, S. Onis et A. Larue : Preprocessing for classification of sparse data : Application to trajectory recognition. *In Proc. IEEE Workshop on Statistical Signal Processing SSP '12*, pages 37–40, 2012.

[MBP+08] J. Mairal, F. Bach, J. Ponce, G. Sapiro et A. Zisserman : Discriminative learned dictionaries for local image analysis. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008*, pages 1–8, 2008.

[MBP+09] J. Mairal, F. Bach, J. Ponce, G. Sapiro et A. Zisserman : Supervised dictionary learning. *In Advances in Neural Information Processing Systems*, pages 1033–1040, 2009.

[MBP12] J. Mairal, F. Bach et J. Ponce : Task-driven dictionary learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34:791–804, 2012.

[MBPS10] J. Mairal, F. Bach, J. Ponce et G. Sapiro : Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010.

[MBSF09] Y. Moudden, J. Bobin, J.-L. Starck et J. Fadili : Dictionary learning with spatio-spectral sparsity constraints. *In Signal Processing with Adaptive Sparse Structured Representations SPARS '09*, 2009.

[MBZJ09] H. Mohimani, M. Babaie-Zadeh et C. Jutten : A fast approach for overcomplete sparse decomposition based on smoothed ℓ_0 norm. *IEEE Trans. on Signal Processing*, 57:289–301, 2009.

[MCW05] D.M. Malioutov, M. Cetin et A.S. Willsky : Homotopy continuation for sparse signal representation. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '05*, pages 733–736, 2005.

[MD10] A. Maleki et D.L. Donoho : Optimally tuned iterative reconstruction algorithms for compressed sensing. *IEEE Journal of Selected Topics in Signal Processing*, 4:330–341, 2010.

[MDMM+05] A. Matysiak, P.J. Durka, E. Martínez-Montes, M. Barwiński, P. Zwoliński, M. Roszkowski et K.J. Blinowska : Time-frequency-space localization of epileptic EEG oscillations. *Acta Neurobiol. Exp.*, 65:435–442, 2005.

[MES08] J. Mairal, M. Elad et G. Sapiro : Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17:53–69, 2008.

[MGB+09] B. Mailhé, R. Gribonval, F. Bimbot, M. Lemay, P. Vandergheynst et J.M. Vesin : Dictionary learning for the sparse modelling of atrial fibrillation in ECG signals. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '09*, pages 465–468, 2009.

[MHA+08] M. Mørup, L.K. Hansen, S.M. Arnfred, L.-H. Lim et K.H. Madsen : Shift-invariant multilinear decomposition of neuroimaging data. *NeuroImage*, 42:1439–1450, 2008.

[MJT11] D.P. Mandic, C. Jahanchahi et C.C. Took : A quaternion gradient operator and its applications. *IEEE Signal Processing Letters*, 18:47–50, 2011.

[MJV+07] G. Monaci, P. Jost, P. Vandergheynst, B. Mailhé, S. Lesage et R. Gribonval : Learning multimodal dictionaries. *IEEE Trans. on Image Processing*, 16:2272–2283, 2007.

[MLBM06] S. Miron, N. Le Bihan et J.I. Mars : Quaternion-MUSIC for vector-sensor array processing. *IEEE Trans. on Signal Processing*, 54:1218–1229, 2006.

[MLG+08] B. Mailhé, S. Lesage, R. Gribonval, F. Bimbot et P. Vandergheynst : Shift-invariant dictionary learning for sparse representations : Extending K-SVD. *In Proc. Eur. Signal Process. Conf. EUSIPCO '08*, 2008.

[MNT04] K. Madsen, H.B. Nielsen et O. Tingleff : Methods for non-linear least squares problems, 2nd edition. Rapport technique, Technical University of Denmark, 2004.

[Moo96] T.K. Moon : The Expectation-Maximization algorithm. *IEEE Signal Processing Magazine*, 13:47–60, 1996.

[MS11] M. Mørup et M. Schmidt : Transformation invariant sparse coding. *In Proc. Machine Learning for Signal Processing MLSP '11*, pages 1–6, 2011.

[MSE03] C.E. Moxey, S.J. Sangwine et T.A. Ell : Hypercomplex correlation techniques for vector images. *IEEE Trans. on Signal Processing*, 51:1941–1953, 2003.

[MSH08] M. Mørup, M.N. Schmidt et L.K. Hansen : Shift invariant sparse coding of image and music data. Rapport technique, Technical University of Denmark, 2008.

[MTS12] F. Meier, E. Theodorou et S. Schaal : Movement segmentation and recognition for imitation learning. *In JMLR W&CP*, volume 22, pages 761–769, 2012.

[MTSS11] F. Meier, E. Theodorou, F. Stulp et S. Schaal : Movement segmentation using a primitive library. *In IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2011*, pages 3407–3412, 2011.

[MVS09] G. Monaci, P. Vandergheynst et F.T. Sommer : Learning bimodal structure in audio-visual data. *IEEE Trans. on Neural Networks*, 20:1898–1910, 2009.

[MZ93] S.G. Mallat et Z. Zhang : Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, 41:3397–3415, 1993.

[NA05] Y. Nishimori et S. Akaho : Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing*, 67:106–135, 2005.

[NFV+12] A. Nonclercq, M. Foulon, D. Verheulpen, C. De Cock, M. Buzatu, P. Mathys et P. Van Bogaert : Cluster-based spike detection algorithm adapts to interpatient and intrapatient variation in spike morphology. *J. Neuroscience Methods*, 210:259–265, 2012.

[NLdS04] E. Niedermeyer et F.H. Lopes da Silva : *Electroencephalography : Basic principles, clinical applications and related fields.* 5th edition, 2004.

[NSL+12a] Q. Noirhomme, A. Soddu, R. Lehembre, M.-A. Bruno, D. Lesenfants, S. Laureys et F. Gomez : A dictionary learning approach to study disorders of consciousness : a preliminary EEG study. *In Belgian Brain Council*, 2012.

[NSL+12b] Q. Noirhomme, A. Soddu, R. Lehembre, M.-A. Bruno, D. Lesenfants, S. Laureys et F. Gómez : Dictionary learning in patients with disorders of consciousness : a preliminary EEG study. *In Human Brain Mapping Conference*, 2012.

[NT09] D. Needell et J.A. Tropp : CoSaMP : Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.*, 26:301–321, 2009.

[NV09] D. Needell et R. Vershynin : Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.*, 9:317–334, 2009.

[NWY11] M.K. Ng, F. Wang et X. Yuan : Inexact alternating direction methods for image recovery. *SIAM J. Sci. Comput.*, 33:1643–1668, 2011.

[OF96] B.A. Olshausen et D.J. Field : Natural image statistics and efficient coding. *Network Comp. Neural Syst.*, 7:333–339, 1996.

[OF97] B.A. Olshausen et D.J. Field : Sparse coding with an overcomplete basis set : a strategy employed by V1 ? *Vision Research*, 37:3311–3325, 1997.

[Ols00] B.A. Olshausen : Sparse coding of time-varying natural images. *In Proc. Int. Conf. Independent Component Analysis and Blind Source Separation*, pages 603–608, 2000.

[Ols01] B.A. Olshausen : Sparse codes and spikes. *In Probabilistic models of the brain : Perception and Neural Function*, pages 257–272. MIT Press, 2001.

[OPT00] M.R. Osborne, B. Presnell et B.A. Turlach : A new approach to variable selection in least squares problems. *IMA Journal of Numerical Analysis*, 20:389–404, 2000.

[PABD06] M.D. Plumbley, S.A. Abdallah, T. Blumensath et M.E. Davies : Sparse representations of polyphonic music. *Signal Process.*, 86:417–431, 2006.

[Plu05] M.D. Plumbley : Geometrical methods for non-negative ICA : Manifolds, Lie groups and toral subalgebras. *Neurocomputing*, 67:161–197, 2005.

[Plu07] M.D. Plumbley : Dictionary learning for l1-exact sparse coding. *In ICA 2007*, volume 4666 de *Lecture Notes in Computer Science*, pages 406–413, 2007.

[PMML95] R.D. Pascual-Marqui, C.M. Michel et D. Lehmann : Segmentation of brain electrical activity into microstates : Model estimation and validation. *IEEE Trans. on Biomedical Engineering*, 42:658–665, 1995.

[PP08] K.B. Petersen et M.S. Pedersen : The matrix cookbook. Rapport technique, Technical University of Denmark, 2008.

[PRK93] Y.C. Pati, R. Rezaiifar et P.S. Krishnaprasad : Orthogonal Matching Pursuit : recursive function approximation with applications to wavelet decomposition. *In Conf. Record of the Asilomar Conf. on Signals, Systems and Comput.*, pages 40–44, 1993.

[PRT12] G.E. Pfander, H. Rauhut et J.A. Tropp : The restricted isometry property for time-frequency structured random matrices. *Probab. Theory Relat. Fields*, 2012.

[PV08] D.S. Pham et S. Venkatesh : Joint learning and dictionary construction for pattern recognition. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008*, pages 1–8, 2008.

[PZA12] M.J. Powell, H. Zhao et A.D. Ames : Motion primitives for human-inspired bipedal robotic locomotion walking and stair climbing. *In Proc. IEEE Int. Conf. Robotics and Automation ICRA*, pages 543–549, 2012.

[Rak11] A. Rakotomamonjy : Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Signal Process.*, 91:1505–1526, 2011.

[RBE10] R. Rubinstein, A.M. Bruckstein et M. Elad : Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98:1045–1057, 2010.

[RJC12] S. Rousseau, C. Jutten et M. Congedo : Closed-looping a P300 BCI using the ErrP. *In Proc. Int. Conf. Neural Computation Theory and Applications NCTA 2012*, pages 732–737, 2012.

[RKD99] B. Rao et K. Kreutz-Delgado : An affine scaling methodology for best basis selection. *IEEE Trans. on Signal Processing*, 47:187–200, 1999.

[RL01] S. Rusinkiewicz et M. Levoy : Efficient variants of the ICP algorithm. *In Proc. Int. Conf. on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

[RNL02] L. Rebollo-Neira et D. Lowe : Optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9:137–140, 2002.

[ROF92] L.I. Rudin, S. Osher et E. Fatemi : Nonlinear total variation based noise removal algorithms. *Physica D : Nonlinear Phenomena*, 60:259–268, 1992.

[RS07] F. Rodriguez et G. Sapiro : Sparse representations for image classification : Learning discriminative and reconstructive non-parametric dictionaries. Rapport technique IMA Preprint 2213, University of Minnesota, December 2007.

[RSAG09] B. Rivet, A. Souloumiac, V. Attina et G. Gibert : xDAWN algorithm to enhance evoked potentials : Application to brain-computer interface. *IEEE Trans. on Biomedical Engineering*, 56: 2035–2043, 2009.

[RSS10] I. Ramirez, P. Sprechmann et G. Sapiro : Classification and clustering via dictionary learning with structured incoherence and shared features. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2010*, pages 3501–3508, 2010.

[Sau02] M.A. Saunders : PDCO : Primal-dual interior-point method for convex objectives. Rapport technique, Stanford University, 2002.

[SBT00] S. Sardy, A.G. Bruce et P. Tseng : Block coordinate relaxation methods for nonparametric wavelet denoising. *Journal of Computational and Graphical Statistics*, 9:361–379, 2000.

[SC95] N. Saito et R.R. Coifman : Local discriminant bases and their applications. *Journal of Mathematical Imaging and Vision*, 5:337–358, 1995.

[SC07] S. Sanei et J. Chambers : *EEG signal processing*. John Wiley & Sons, 2007.

[Sch66] P.H. Schonemann : A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31:1–10, 1966.

[SE10] K. Skretting et K. Engan : Recursive least squares dictionary learning algorithm. *IEEE Trans. on Signal Processing*, 58:2121–2130, 2010.

[SED04] J.-L. Starck, M. Elad et D.L. Donoho : Redundant multiscale transforms and their application for morphological component analysis. *Advances in Imaging and Electron Physics*, 132:287–348, 2004.

[SH06] K. Skretting et J.H. Husøy : Texture classification using sparse frame based representation. *EURASIP J. on Applied Signal Processsing*, 2006:1–11, 2006.

[SHA06] K. Skretting, J.H. Husøy et S.O. Aase : General design algorithm for sparse frame expansions. *Signal Process.*, 86:117–126, 2006.

[SIBD11] C. Soussen, J. Idier, D. Brie et J. Duan : From Bernoulli-Gaussian deconvolution to sparse signal restoration. *IEEE Trans. on Signal Processing*, 59:4572–4584, 2011.

[SJS08] R. Sameni, C. Jutten et M.B. Shamsollahi : Multichannel electrocardiogram decomposition using periodic component analysis. *IEEE Trans. on Biomedical Engineering*, 55:1935–1940, 2008.

[SKM+09a] C. Sielużycki, R. König, A. Matysiak, R. Kuś, D. Ircha et P.J. Durka : Single-trial evoked brain responses modeled by multivariate matching pursuit. *IEEE Trans. on Biomedical Engineering*, 56:74–82, 2009.

[SKM+09b] C. Sielużycki, R. Kuś, A. Matysiak, P.J. Durka et R. König : Multivariate matching pursuit in the analysis of single-trial latency of the auditory M100 acquired with MEG. *Int. Journal of Bioelectromagnetism*, 11:155–160, 2009.

[SL05a] E. Smith et M.S. Lewicki : Efficient coding of time-relative structure using spikes. *Neural Comput.*, 17:19–45, 2005.

[SL05b] E. Smith et M.S. Lewicki : Learning efficient auditory codes using spikes predicts cochlear filters. *In Advances in Neural Information Processing Systems NIPS*, pages 1289–1296, 2005.

[SL06] E. Smith et M.S. Lewicki : Efficient auditory coding. *Nature*, 439:978–982, 2006.

[SLY10] Y. Su, L. Liu et Y. Yang : Optimal trajectory space finding for nonrigid structure from motion. *In Advanced Concepts for Intelligent Vision Systems ACIVS '10*, volume 6474 de *Lecture Notes in Computer Science*, pages 357–366, 2010.

[SP11] S. Scholler et H. Purwins : Sparse approximations for drum sound classification. *IEEE Journal of Selected Topics in Signal Processing*, 5:933–940, 2011.

[SS85] J.T. Schwartz et M. Sharir : Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. Rapport technique Robotics Report 46, Courant Institute of Mathematical Sciences, New York University, 1985.

[SS09] A. Szlam et G. Sapiro : Discriminative k-metrics. *In Proc. Int. Conf. Machine Learning ICML '09*, pages 1009–1016, 2009.

[SS10] P. Sprechmann et G. Sapiro : Dictionary learning and sparse coding for unsupervised clustering. *In Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP 2010*, pages 2042–2045, 2010.

[SSAS11] S. Satpathy, L. Sharma, A.K. Akasapu et N. Sharma : Towards mining approaches for trajectory data. *Int. J. of Advances in Science and Technology*, 2:38–43, 2011.

[SSD03] D.J. Strauss, G. Steidl et W. Delb : Feature extraction by shape-adapted local discriminant bases. *Signal Process.*, 83:359–376, 2003.

[TCA11] D. Tlalolini, C. Chevallereau et Y. Aoustin : Human-like walking : Optimal motion of a bipedal robot with toe-rotation motion. *IEEE/ASME Trans. on Mechatronics*, 16:310–320, 2011.

[TF11a] I. Tošić et P. Frossard : Dictionary learning. *IEEE Signal Processing Magazine*, 28:27–38, 2011.

[TF11b] I. Tošić et P. Frossard : Dictionary learning for stereo image representation. *IEEE Trans. on Image Processing*, 20:921–934, 2011.

[TGS06] J.A. Tropp, A.C. Gilbert et M.J. Strauss : Algorithms for simultaneous sparse approximation ; Part I : Greedy pursuit. *Signal Process. - Sparse approximations in signal and image processing*, 86:572–588, 2006.

[THB04] L. Torresani, A. Hertzmann et C. Bregler : Learning non-rigid 3D shape from 2D motion. *In Advances in Neural Information Processing Systems NIPS '04*, pages 1555–1562, 2004.

[THB08] L. Torresani, A. Hertzmann et C. Bregler : Nonrigid structure-from-motion : Estimating shape and motion with hierarchical priors. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30:878–892, 2008.

[Tib96] R. Tibshirani : Regression shrinkage and selection via the LASSO. *J. R. Statist. Soc. B*, 58:267–288, 1996.

[TM10] C.C. Took et D.P. Mandic : Quaternion-valued stochastic gradient-based adaptive IIR filtering. *IEEE Trans. on Signal Processing*, 58:3895–3901, 2010.

[TMA+12] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K.J. Miller, G. Müller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert et B. Blankertz : Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6:1–31, 2012.

[Tro04] J.A. Tropp : Greed is good : algorithmic results for sparse approximation. *IEEE Trans. on Information Theory*, 50:2231–2242, 2004.

[Tro06a] J.A. Tropp : Algorithms for simultaneous sparse approximation ; Part II : Convex relaxation. *Signal Process. - Sparse approximations in signal and image processing*, 86:589–602, 2006.

[Tro06b] J.A. Tropp : Just relax : Convex programming methods for subset selection and sparse approximation. *IEEE Trans. on Information Theory*, 52:1030–1051, 2006.

[TW10] J.A. Tropp et S.J. Wright : Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98:948–958, 2010.

[TWH13] D.E. Thompson, S. Warschausky et J.E. Huggins : Classifier-based latency estimation : a novel way to estimate and predict BCI accuracy. *Journal of Neural Engineering*, 10:1–7, 2013.

[Ume91] S. Umeyama : Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:376–380, 1991.

[VB02] P. Vincent et Y. Bengio : Kernel matching pursuit. *Machine Learning*, 48:165–187, 2002.

[VEG12] C. Vollmer, J.P. Eggert et H.-M. Gross : Modeling human motion trajectories by sparse activation of motion primitives learned from unpartitioned data. *In KI 2012 : Advances in Artificial Intelligence*, volume 7526 de *Lecture Notes in Computer Science*, pages 168–179, 2012.

[VGD04] M. Vlachos, D. Gunopulos et G. Das : Rotation invariant distance measures for trajectories. *In Proc. SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 707–712, 2004.

[VML04] V.D. Vrabie, J.I. Mars et J.-L. Lacoume : Modified singular value decomposition by means of independent component analysis. *Signal Process.*, 84:645–652, 2004.

[WEK03] H. Wersing, J. Eggert et E. Körner : Sparse coding with invariance constraints. *In Proc. Int. Conf. Artificial Neural Networks ICANN*, pages 385–392, 2003.

[WG11] W. Wu et S. Gao : Learning event-related potentials (ERPs) from multichannel EEG recordings : a spatio-temporal modeling framework with a fast estimation algorithm. *In Int. Conf. IEEE EMBS*, pages 6959–6962, 2011.

[WH60] B. Widrow et M.E. Hoff : Adaptive switching circuits. *In Proc. WESCON Conv. Rec.*, pages 96–104, 1960.

[Wid71] B. Widrow : *Adaptive Filters*, pages 563–586. Aspects of Network and System Theory, 1971.

[WNF09] S.J. Wright, R.D. Nowak et M.A.T. Figueiredo : Sparse reconstruction by separable approximation. *IEEE Trans. on Signal Processing*, 57:2479–2493, 2009.

[WSV91] M.W. Walker, L. Shao et R.A. Volz : Estimating 3-D location parameters using dual number quaternions. *CVGIP : Image Understanding*, 53:358–367, 1991.

[WTS07] B.H. Williams, M. Toussaint et A.J. Storkey : A primitive based generative model to infer timing information in unpartitioned handwriting data. *In Proc. Int. Joint Conf. on Artificial Intelligence IJCAI*, pages 1119–1124, 2007.

[WTS08] B.H. Williams, M. Toussaint et A.J. Storkey : Modelling motion primitives and their timing in biologically executed movements. *In Advances in Neural Information Processing Systems NIPS*, volume 20, pages 1609–1616, 2008.

[WYG+09] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry et Y. Ma : Robust face recognition via sparse representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31:210–227, 2009.

[XSL+09] L. Xu, P. Stoica, J. Li, S.L. Bressler, X. Shao et M. Ding : ASEO : A method for the simultaneous estimation of single-trial event-related potentials and ongoing brain activities. *IEEE Trans. on Biomedical Engineering*, 56:111–121, 2009.

[YBD09] M. Yaghoobi, T. Blumensath et M.E. Davies : Dictionary learning for sparse approximations with the majorization method. *IEEE Trans. on Signal Processing*, 57:2178–2191, 2009.

[YBW85] R. Yarlagadda, J.B. Bednar et T.L. Watt : Fast algorithms for ℓ_p deconvolution. *IEEE Trans. on Accoust., Speech, and Signal Processing*, ASSP-33:174–182, 1985.

[YDD09] M. Yaghoobi, L. Daudet et M.E. Davies : Parametric dictionary design for sparse coding. *IEEE Trans. on Signal Processing*, 57:4800–4810, 2009.

[YR11] F. Yger et A. Rakotomamonjy : Wavelet kernel learning. *Pattern Recognition*, 44:2614–2629, 2011.

[YWHM10] J. Yang, J. Wright, T.S. Huang et Y. Ma : Image super-resolution via sparse representation. *IEEE Trans. on Image Processing*, 19:2861–2873, 2010.

[YZ11] J. Yang et Y. Zhang : Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM J. Sci. Comput.*, 33:250–278, 2011.

[YZFZ11] M. Yang, L. Zhang, X. Feng et D. Zhang : Fisher discrimination dictionary learning for sparse representation. *In Proc. Int. Conf. Computer Vision ICCV '11*, pages 543–550, 2011.

[YZYZ10] M. Yang, L. Zhang, J. Yang et D. Zhang : Metaface learning for sparse representation based face recognition. *In Proc. IEEE Int. Conf. Image Processing ICIP 2010*, pages 1601–1604, 2010.

[ZEP12] R. Zeyde, M. Elad et M. Protter : On single image scale-up using sparse-representations. *In Curves and Surfaces*, volume 6920 de *Lecture Notes in Computer Science*, pages 711–730, 2012.

[Zha97] F. Zhang : Quaternions and matrices of quaternions. *Linear Algebra and its Applications*, 251:21–57, 1997.

[ZL10] Q. Zhang et B. Li : Discriminative K-SVD for dictionary learning in face recognition. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2010*, pages 2691–2698, 2010.

[Zou05] T. Zou, H. Hastie : Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, 67:301–320, 2005.

[ZSD12] M. Zhu, H. Sun et Z. Deng : Quaternion space sparse decomposition for motion compression and retrieval. *In Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '12, pages 183–192, 2012.

[ZZZ+12] S. Zhang, Y. Zhan, Y. Zhou, M.G. Uzunbas et D.N. Metaxas : Shape prior modeling using sparse representation and online dictionary learning. *In Medical Image Computing and Computer-Assisted Intervention - MICCAI 2012*, volume 7512 de *Lecture Notes in Computer Science*, pages 435–442, 2012.

Publications

Articles journaux

Q. Barthélemy, C. Gouy-Pailler, Y. Isaac, A. Souloumiac, A. Larue et J.I. Mars, Multivariate temporal dictionary learning for EEG, *Journal of Neuroscience Methods*, vol. 215, pp. 19-28, 2013.

Q. Barthélemy, A. Larue, A. Mayoue, D. Mercier et J. I. Mars, Shift & 2D rotation invariant sparse coding for multivariate signals, *IEEE Trans. on Signal Processing*, vol. 60, pp. 1597-1611, 2012.

Articles conférences

Y. Isaac, Q. Barthélemy, J. Atif, C. Gouy-Pailler et M. Sebag, Multi-dimensional sparse structured signal approximation using split Bregman iterations, In *IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '13*, pp. 3826-3830, 2013.

Q. Barthélemy, A. Larue et J. I. Mars, 3D rotation invariant decomposition of motion signals, In *Computer Vision - ECCV 2012*, *Lecture Notes in Computer Science* 7585, pp. 172-182, 2012.

A. Mayoue, Q. Barthélemy, S. Onis et A. Larue, Preprocessing for classification of sparse data : application to trajectory recognition, In *Proc. IEEE Workshop on Statistical Signal Processing SSP '12*, pp. 37-40, 2012.

Q. Barthélemy, A. Larue et J. I. Mars, Quaternionic sparse approximation, In *Conf. on Applied Geometric Algebras in Computer Science and Engineering AGACSE*, 2012.

A. Mouraud, Q. Barthélemy, A. Mayoue, C. Gouy-Pailler, A. Larue et H. Paugam-Moisy, From neuronal cost-based metrics to sparse coded signal classification, In *Proc. Eur. Symp. on Artificial Neural Networks, Computational Intelligence and Machine Learning ESANN*, pp. 311-316, 2012.

Q. Barthélemy, A. Larue, A. Mayoue, D. Mercier et J. I. Mars, Multivariate dictionary learning and shift & 2D rotation invariant sparse coding, In *Proc. IEEE Workshop on Statistical Signal Processing SSP '11*, pp. 645-648, 2011.

Articles conférences nationales

Q. Barthélemy, M. Sangnier, A. Larue et J. I. Mars, Comparaison de descripteurs pour la classification de décompositions parcimonieuses invariantes par translation, In *XXIV Colloque GRETSI - Traitement du Signal et des Images*, 2013.

Y. Isaac, Q. Barthélemy, J. Atif, C. Gouy-Pailler et M. Sebag, Régularisations spatiales pour la décomposition de signaux EEG sur un dictionnaire temps-fréquence, In *XXIV Colloque GRETSI - Traitement du Signal et des Images*, 2013.

Q. Barthélemy, A. Larue, A. Mayoue, D. Mercier et J. I. Mars, Apprentissage de dictionnaires multivariés et décomposition parcimonieuse invariante par translation et par rotation 2D, In *XXIII Colloque GRETSI - Traitement du Signal et des Images*, 2011.

Articles soumis

S. Chevallier, Q. Barthélemy et J. Atif, Metrics for multivariate dictionaries, Preprint arXiv :11302.4242.

Représentations parcimonieuses pour les signaux multivariés

Résumé

Dans cette thèse, nous étudions les méthodes de décomposition de signaux et d'apprentissage de dictionnaire qui fournissent des représentations parcimonieuses. Ces méthodes permettent d'analyser des bases de données très redondantes à l'aide de dictionnaires d'atomes appris, avec des performances supérieures à celles des dictionnaires analytiques classiques. Nous considérons plus particulièrement des signaux multivariés résultant de l'acquisition simultanée de plusieurs grandeurs, comme les signaux EEG ou les signaux de mouvements 2D et 3D. Nous étendons les méthodes de représentations parcimonieuses au modèle multivarié, pour prendre en compte les interactions entre les différentes composantes acquises simultanément. Ce modèle est plus flexible que l'habituel modèle multicanal qui impose une hypothèse de rang 1. Nous étudions des modèles de représentations invariantes : invariance par translation temporelle, invariance par rotation, etc. En ajoutant des degrés de liberté supplémentaires, chaque noyau est potentiellement démultiplié en une famille d'atomes, translatés à tous les échantillons, tournés dans toutes les orientations, etc. pour engendrer un dictionnaire d'atomes très redondant. Les méthodes de décomposition et d'apprentissage de dictionnaire sont adaptées à ces modèles. Dans le cas de l'invariance par rotation 2D et 3D, nous constatons l'efficacité de l'approche non-orientée sur celle orientée, même dans le cas où les données ne sont pas tournées. En effet, le modèle non-orienté permet de détecter les invariants des données et assure la robustesse à la rotation quand les données tournent. Nous constatons aussi la reproductibilité des décompositions parcimonieuses sur un dictionnaire appris. Dans le cas de l'invariance par translation, des vecteurs de caractéristiques sont extraits des décompositions grâce à des fonctions de groupement consistantes par translation. La classification résultante est robuste à la translation des signaux.

Mots clés : parcimonie, approximation parcimonieuse, apprentissage de dictionnaire, signaux multivariés, invariances.

Sparse representations for multivariate signals

Abstract

In this thesis, we study signals decomposition and dictionary learning methods which provide sparse representations. These methods allow to analyze very redundant data-bases thanks to learned atoms dictionaries, which are more efficient in representation quality than classical analytic dictionaries. We more particularly consider multivariate signals coming from the simultaneous acquisition of several quantities, as EEG signals or 2D and 3D motion signals. We extend sparse representation methods to the multivariate model, in order to take into account interactions between the different components acquired simultaneously. This model is more flexible than the common multichannel one which imposes a hypothesis of rank 1. We study models of invariant representations : invariance to temporal shift, invariance to rotation, etc. Adding supplementary degrees of freedom, each kernel is potentially replicated in an atoms family, translated at all samples, rotated at all orientations, etc. to generate a very redundant atoms dictionary. Decomposition and dictionary learning methods are adapted to these models. In the 2D and 3D rotation invariant case, we observe the efficiency of the non-oriented approach over the oriented one, even when data are not revolved. Indeed, the non-oriented model allows to detect data invariants and assures the robustness to rotation when data are revolved. We also observe the reproducibility of the sparse decompositions on a learned dictionary. In the shift-invariance case, features are extracted from decompositions thanks to shift consistent pooling functions. The resulting classification is robust to signals translation.

Key words : sparsity, sparse approximation, dictionary learning, multivariate signals, invariances.